HARDWARE TESTBED EXPERIENCE IN AUTOMATED ASSEMBLY OF SPACE STRUCTURES

[illegible]

Will, Ralph

Ralph Will, Marvin Rhodes, William Doggett, Catherine Herstrom,
Carolyn Grantham, Cheryl Allen, Daniel Sydow, Eric Cooper
NASA Langley Research Center

Coung Quach, and Marion Wise
Lockheed Engineering and Sciences Company, Inc.

Published by Kluwer Academic Publishers as part of the test Intelligent Robotic Systems for Space Exploration. Coordinated by Rensselaer Polytechnic Institute's Center for intelligent Robotics for Space Exploration (CIRSSE), Troy, New York

AN AUTOMATED ASSEMBLY SYSTEM FOR LARGE SPACE STRUCTURES

Ralph Will
Marvin Rhodes
Billy Doggett
Catherine Herstrom
Carolyn Grantham
Cheryl Allen
Dan Sydow
Eric Cooper
Patrick Quach
Jim Wise

PREFACE

This chapter presents a research program that is being conducted at the NASA Langley Research Center to evaluate methods for automated telerobotic assembly of large space truss structures. The research program, facility, and hardware are described and a summary of the results obtained to date is presented. The research is intended to identify the specific problems and considerations which must be dealt with when the total assembly process is addressed in detail. The assembly operation is structured to focus on a generic truss structure rather than on a particular mission or design; however, the assembly issues being investigated are typical of many automated in-space processes. The program was initiated to study automated truss assembly operations and no other objectives were included in this initial investigation. The structure, although not space qualified, was designed to be representative of that required for future antennas and observatories.

Implementation of this research effort has required the integration of a strong interdisciplinary team composed of specialists in mechanical and structural design concepts, automation and robotics, controls, software development, configuration management, and electronic design and development. Many support personnel have contributed to the success of this research effort by their dedication and hard work. A well defined focus has enhanced the capability of the team to work together and meet the goals and milestones in a timely manner.

INTRODUCTION

A number of proposed space missions both to and from planet Earth require large truss structures to provide a stiff and stable platform for experimental measurements, observation antennas/telescopes and habitats/shelters. An example of a planned mission to planet Earth is shown in figure 1(a) and an example of a mission from planet Earth is shown in figure 1(b); figure 1(a) is a large antenna, and figure 1(b) is an aerobrake concept for a Mars mission vehicle. The truss structures for these, as well as other future missions may involve the assembly of thousands of members. Recent studies (Refs. 1 and 2) have elevated the concern regarding the use of astronauts to perform inspace construction operations. Therefore, alternative assembly techniques to those that traditionally rely on astronaut Extra-Vehicular Activity (EVA) must be explored. The reference studies have also recommended that astronaut Intra-Vehicular Activity (IVA) time be minimized, which may limit the use of astronauts as teleoperators for assembly operations that may involve many hours of repetitive operations. One attractive alternative is to design an automated telerobotic system that can utilize either an astronaut or earth based operator as an executive monitor who is called upon to intervene only when the automated system encounters a problem and requires assistance. This mode of operation, known as "supervised autonomy", holds the most promise for the accomplishment of large or complex assembly and construction tasks with the limited crew resources available on orbit. Supervised autonomy has the additional advantage that operations can be monitored from the ground if they involve no time critical control functions.

To date, very little work has been directed toward interdisciplinary research to develop automated robotic assembly methods for large erectable truss structures. The current program was initiated within the past several years at the NASA Langley Research Center by merging basic technology in robotics and truss structure design and assembly. The program focuses on the actual automated assembly of a generic structural unit that serves as the basic element for very large structural systems. Specific objectives of the program are to determine what types of joining and end-effector mechanisms are suitable for telerobotic operation, to develop a software architecture capable of reliably performing a complex assembly task that incorporates realistic system errors, and to provide an operator interface compatible with the volume of internal information necessary for successful operation of the automated system. This effort provides practical experience in the assembly of truss hardware designed for automated telerobotic operations. An additional objective is to collect information on actual assembly operations to develop time lines characterizing automated telerobotic truss assembly and construction. It also has the potential to provide an operational testbed for more advanced assembly techniques such as automated path and sequence planners and machine vision guidance operations. The hardware testbed can also be used to measure the effectiveness of the operator interface and display system in communicating with operators who have only modest levels of training.

This chapter describes the system design considerations, facility and system hardware, and the software and operator interface functions involved in implementing the hardware testbed. The test facility is operational and the system executive program and error recovery software is performing all assembly operations in a fully automated mode. Several end-to-end assembly and disassembly sequences for the complete 102-member truss structure have been performed and information on timing, error frequency, error causes, and recovery techniques have been collected. The sequence for the assembly tests conducted to date will be discussed along with the research directions toward a more robust system.

FACILITY AND HARDWARE DESCRIPTION

At the initiation of this program, several ground rules were established to guide the development of the component design. These ground rules were: 1) existing "off-the-shelf" components would be used directly, or modified for use, where possible to hasten implementation; 2) design and fabrication would incorporate inexpensive materials and simple systems so that they could be modified easily as experience and future research requirements dictate; 3) passive guidance features would be incorporated wherever possible to aid alignment for minimizing position errors; 4) all assembly operations would be reversible so that truss repair and error recovery procedures could also be automated and; 5) system hardware and software design would be based on total system automation with operator intervention rather than enhancing a teleoperated system with automated functions. The current automated assembly facility is shown schematically in figure 2(a) and a photograph of the actual hardware system is shown in figure 2(b). Important aspects of the various components are discussed in subsequent sections and a detailed description of the performance characteristics is presented in reference 3.

Robot and Motion Bases

The robot arm shown in figure 2 is a six-degree-of-freedom, electrically driven, industrial robot selected for its reach envelope (1.52 m (5 feet)), payload capacity 89N (20 lbf.), positioning repeatability (±0.010 cm (0.004 in.)), and reliability. No modifications have been made to the robot over those supplied as options by the manufacturer. The robot is mounted on an x-y Cartesian translational motion base that has approximately 6.1 m (20 feet) of travel in both x and y directions and has a measured positioning accuracy of ±0.005 cm (0.002 in.) using linear encoders. The truss structure is assembled on a rotating motion base with ± 3 revolutions from the nominal reference location and a positioning accuracy of ±0.25 mm (0.01in.) at a radial distance of 6.1 m (20 ft). The motion bases are designed to minimize deflections that induce position errors due to the increasing mass of the truss during assembly and repositioning of the robot. Motion base drive motors on all three axes are controlled by an 80286 microprocessor-based personal computer. The robot computer is based on an M-68000 microprocessor and all robot motions are programmed in a modified Basic programming language supplied by the manufacturer.

Truss and Joining Mechanism

The truss structure selected for this study is a regular tetrahedral configuration shown by the model in figure 3(a). This configuration was chosen for several reasons: 1) it is representative of supporting trusses required for large antennas, reflectors, and aerobrakes; 2) it is comprised of regular hexagonal rings as shown in the figure, therefore, the developed assembly procedure can be applied to construct large multi-ring units; and 3) the truss has some special geometric characteristics that provide desirable access paths for the installation of struts. The truss hardware used in this investigation has members that are nominally 2 meters (6.6 feet) long and 2.64 cm (1.04 in) in diameter. The 2-ring test model has a diameter of 8 meters (26.25 ft) and the complete structure consists of 102 struts and 31 nodes.

The special geometric characteristics noted previously can be seen in figure 3(b). The truss geometry includes a natural set of orthogonal axes and there are sets of planes normal to this axis system. Every member in the truss is oriented to lie within one of the sets of planes and the members in a given plane form square geometric patterns (fig. 3(b)). The squares provide the maximum area for collision-free manipulation and access for installation of strut members. The squares and their planar orientation are referred to here after as the member's insertion plane. Assembly of the truss involves maneuvering the strut along a predefined path into the insertion plane, aligning it parallel to the installed orientation approximately 10 cm (4 in.) in front of the installed position, and then moving it directly to the installation position. The assembly process begins by connecting members to three pre-mounted nodes on the rotating motion base to build the first hexagonal ring. Additional detail on assembly operations will be presented in a subsequent section.

The joint connectors for space truss structures have a number of requirements that must be met for the truss to be assembled and have a predictable performance. The connectors must have a predictable linear load-deflection response for reverse tension-compression load cycles and have no "free-play". The joints must permit the strut members to be installed by side entry between two nodes a fixed distance apart, because the truss is structurally redundant and it is desirable to minimize disassembly to replace or repair a damaged member. Structural redundancy also requires the length of the member to be accurately set and ramped entry guides on the joint are desirable to aid the insertion operation. Additional requirements are defined in reference 3.

The truss joints and nodes developed for this test assembly are shown in figure 4. Each node must be capable of connecting a total of nine struts that are aligned with the node center; six of the connecting struts lie in the plane of the face and the remaining three are core members that connect the top and bottom faces of the truss. The joint is composed of two parts, a connector section which is bonded to the graphite epoxy tube to form a strut and the receptacle section which is mechanically attached to the node. The connector section is inserted into the node-mounted receptacle and a locking nut is turned to draw the connector plunger toward the strut, securing the strut to the receptacle. The locking mechanism applies approximately 890 N (200 lbf) preload to the joint connection to eliminate "free-play". The joints are fabricated from aluminum and have a mass of 134g (0.296 lbm). Additional details concerning the mechanism and its operation can be found in reference 4.

Following fabrication of the truss struts and node components, they were assembled and measured to evaluate the accuracy of the total truss structure. The truss was manually assembled for this series of tests with the design torque closure limits applied to the joints. Three sets of photogrammetry measurements were made on a set of high contrast retro-reflective targets centrally positioned on each of the 19 upper surface nodes. The test measurements indicated that the rms deviation between the 19 upper surface nodes and the best fit plane was about 0.015 cm (0.006 in.) and the rms positioning repeatability for two assembly tests was about 0.005 cm (0.002) in. Additional information regarding these tests is presented in reference 4.

End-Effector

At the start of the program, a number of truss joints were evaluated for possible implementation and/or adaptation to the automation process. The current joint was selected because, in addition to the structural capabilities noted above, it could be adapted to operate with a simple end-effector. The development of the joint and end-effector as a total design aided in the development and success of the project. The end-effector is a specialized tool that performs all operations required for strut installation or removal including the locking of the joint connector. The end-effector's operational functions are shown schematically in figure 5(a) and a photograph of the hardware mechanisms is shown in figure 5(b). The strut is grasped by a set of strut holders which close around a hexagonal alignment and grasp adapter that is bonded to the strut tube (fig. 4). The hexagonal shape aids in circumferentially aligning the strut with the end-effector and is a configuration that can be easily grasped by the end-effector jaws. The center of the hexagonal adapter has a machined vee-groove that mates with a protrusion in the end-effector jaws to passively position the strut axially in the end-effector. These passive alignment features are representative of the many such details that are incorporated in all of the components where alignment is critical for connecting or mating.

To insert a strut in the truss, the end-effector is moved to a predetermined ("taught") position in the insertion plane. At this location the fingers are closed on the receptacle. The fingers are configured so that during closure they are passively aligned with the receptacle if the receptacle is at any location within a cylindrical envelope 5 cm (2 in.) in dia. by 1.5 cm (0.6) long. Both ends of the strut are pushed forward by platforms, inserting the connector into the receptacle. The strut is held in place while small gear-head motors lock the joints. The strut holders are then unlatched, the platforms are retracted, and the receptacle fingers are opened to complete the installation. The end-effector is designed to permit operation either with a node preattached to either end of the strut or to insert a strut into two nodes already in the structure.

All end-effector mechanisms and actuators are equipped with simple sensors such as microswitches or proximity probes so that a computer program can monitor the operation and notify the operator if a problem occurs. Small video cameras (fig. 5(b)) are mounted on each end of the end-effector to permit operator monitoring of end-effector mechanisms and to verify the sensor information. The total mass of the end-effector including a strut is about 6 kg(13.5 lbm). A commercial force/torque load cell is mounted between the end-effector and the robot wrist to provide compliant move capability during both strut pick-up and installation operations. This aspect of the assembly operation will be discussed in a subsequent section.

Strut Storage

The requirements defined for storage of the struts in preparation for truss assembly included the following: (1) all struts be placed at a convenient location within the reach envelope of the robot without interfering with other assembly operations, (2) the precise location and orientation of each strut be fixed, (3) the struts be packaged in the smallest volume possible and the individual members still be readily accessible, (4) each member be restrained but capable of being removed or installed in the storage container by a small latching force. Several concepts were initially considered including a device that would present each strut to the end-effector at a common location. Implementing this concept appeared difficult and the packing efficiency appeared to be poor. An alternative concept, having an additional advantage of being passive, was developed which consists of a series of trays stacked in a rack behind the robot arm (fig. 2(a)). The trays, shown in figure 6, are aluminum frames which hold thirteen struts each, between vertical positioning pins. The positioning pins have spring loaded pin plungers on both sides to hold the struts in place. A small force is required to extract each strut from its storage slot. Alignment and grasp adapters like those used to pick up the strut with the end-effector are used to interface with the positioning pin. The flat sides on these hexagonal adapters are trapped between the tray above and the base of the holding tray to prevent the struts from rotating. Nodes are preattached to selected struts in accordance with the assembly sequence. The protrusion of the receptacles from the preattached nodes imposes a sizable storage space requirement and a special stacking arrangement had to be devised to coordinate with the assembly scenario. The stacking arrangement is shown in the sketch of figure 7. The arrangement has 4 prenoded struts in tray 1 at the top of the stack. The prenoded struts are located in the first and last tray slots with two other noded struts equally spaced in-between separated by 3 unnoded struts. The second tray also has 4 prenoded struts at the same slot locations, however, the nodes are all placed in the opposite end of the tray to those of tray 1. Tray 3 has 3 prenoded struts with the nodes staggered to fit in slots between those of tray 1. Tray 4 is similar to tray 3 except that the nodes are on the same end of the tray as those of tray 2. Tray 5 is identical to tray 1 and the four tray pattern is repeated. Therefore, all even numbered trays have struts with alternating nodes of 4 and 3 on the same tray end, and likewise with the odd numbered trays. Due to the efficient packaging of struts in the trays, the receptacle fingers on the end-effector must be closed to pick up an unnoded strut so as not to interfere with the receptacle on adjacent prenoded struts. The complete 102 member truss is packaged in 9 trays with several positions in selected trays left empty to accommodate the assembly sequence. Each tray has cylindrical handles on each end with hexagonal positioning and alignment adaptors identical to those on the struts. The trays are picked up by the end-effector as they are emptied and transferred to a storage canister located to the left side of the robot arm (see figure 2).

System Control and Communication

The entire system is managed by several digital computers serially connected through RS-232 communication lines as shown schematically in figure 8. Overall system executive and operator interface functions are performed on a micro-VAX workstation and implemented in FORTRAN. Also located on the VAX are the robot arm path control logic and the assembly system error recovery algorithms. The robot carriages are controlled by an indexer board on an 80286 personal computer (PC). Commands to this board are generated by a driver program on the PC written in BASIC. The robot arm motions are controlled by a program written in a modified BASIC programming language developed by the robot manufacturer. The robot program is executed on the robot's M-68000 processor in response to commands from the executive program. The robot processor also includes local data which defines pretaught paths from the canister to the installation position of each strut. Storing the local data in the robot processor minimizes the information that must be transferred between processors during execution.

Operator Control

The operator's workstation is shown in figure 9. The operator has four basic sources of information available to monitor system operation. These are 1) the computer CRT menu display, 2) a video system display, 3) the CRT display from the robot computer and, 4) a panoramic view through the control room windows. All operator command inputs are entered via a conventional terminal keyboard. The operator's role is to initiate and monitor the automated assembly process using a series of hierarchical menus which are displayed dynamically. The automated system is directed by a command file to install struts in a predetermined order. The operator monitors all operations via the displayed menus which highlight the current system command status. If an error occurs a brief summary of the error condition and a menu of operator interventions for recovery from the error are displayed. The operator uses the video displays to verify the nature and determine the severity of the problem. All error recovery interventions consist of keyboard commands from menu selections. The assembly will not proceed until the error is corrected. If the error cannot be resolved, the system reverses the sequence and rolls back to the state where the last command file directive was issued. Additional discussion of the operator's display and procedures is provided in Section V, Software Structure.

The video system is comprised of four fixed position cameras: one located on each end of the end-effector to provide a close-up view of the end-effector mechanisms; one located on the Y carriage behind the manipulator arm to provide an over-the-shoulder view of the strut installation and a direct overhead view of canister operations; and one camera fixed to a wall approximately 6.1 m (20 ft.) high and 12.2 m (40 ft.) from the rotational motion base to provide surveillance of assembly operations from the right rear quarter of the robot. Both the over-the-shoulder and wall mounted cameras are equipped with pan/tilt/zoom capability which is controlled by joysticks from the operator's workstation.

The CRT display for the robot computer provides the operator with a visual cue of the robot commands and states. This display provides the operator backup information to the menus associated with the executive program to aid in assessing error sources and recovery operations that involve robot motion. Also, the forces and moments measured using the force/torque load cell are displayed on the robot CRT display.

The panoramic view that the operator has through the control room window does not provide substantial information to correct detail errors because the control room is approximately 10.7 m (35 ft.) from the turntable-- too far to observe mechanism operations. The view is conveniently located to permit the detection of potential collisions between the motion bases and the truss. Tests during which blinds were drawn over these windows, however, resulted in no adverse effects on operator performance or comfort.

ASSEMBLY OPERATIONS

Each strut in the truss assembly has a specific name based on an identification convention developed for this investigation. When a strut is selected for installation, the system data base is checked to establish the status of the requested strut and determine if the strut is being installed in an acceptable order. A number of factors may affect the order. For example, connecting struts required to support the member must have been previously installed and a collision free path for installation must be available. If all factors are satisfied, the installation sequence is initiated. The first operation of an installation sequence is to acquire the strut from the canister tray and hold the strut with the arm in a rest position just above the canister. The motion bases are positioned so that the arm can move the strut along a predefined collision-free path to the installation point in the insertion plane. The end-effector inserts the strut connectors in the joint receptacles and locks the joint connector. The arm then returns to the rest position over the canister. The following subsections give a detailed account of each of these operations and outline some of the checks and procedures that are followed if an error is detected. The initial assembly sequence developed for this investigation, including the rules that were followed during the development, is listed in the Appendices A and B.

Strut Identification Convention

Since the tetrahedral truss has a regular geometric pattern composed of basic subelements, a naming convention was desired that would be common for those struts with similar orientations and positions. A convention that was suitable for any "n" ring configuration was also desirable. A sketch illustrating the adopted convention is shown in figure 10. The figure shows a top planform view of a large planar truss. The struts in the top face are represented by lines of medium width, the struts in the bottom face are represented by lines of narrow width and the core struts that connect the top and bottom faces are represented by the dashed lines. The wide solid lines outline the "n" hexagonal rings which are used in this paper to describe the truss. Each ring is composed of 6(n-1)+3 cell units and the members within each cell are defined by their end positions that fall on the even numbered locations of a conventional clock face, noted on the figure as the operator's convention. Any strut in the truss can therefore be specified by its ring number, cell number and clock position. An example of this notation used for a member of ring 3 and cell 1 whose ends are at clock positions 10 and 2 is R3 C1/10_2. Using this convention, all top and bottom face struts have a unique designation. However, since the core struts lie on the boundary of the individual cells, they can be identified by two cell designations.

The naming convention defined above permits an operator to locate any member in a large multi-member truss with respect to an origin located at a specific reference node. The convention is based on a fixed orientation of the operator and the truss. The reference node for the automated assembly experiment is the pivot node on the axis of the turntable. Since end positions of the struts change with the angular orientation of the turntable, the designation of a member from the vantage point of the robot may be different than the designation from the vantage point of the operator. From the vantage point of the robot, the truss contains 12 unique positions if the turntable is positioned in 120 degree increments.

Strut Pick-up From the Canister

Acquiring a strut from the canister (and strut return to the canister during disassembly) is accomplished using information from a system data base described in Section IV. This information includes the strut's assigned tray number and the slot number within the tray. The data base also contains information regarding preattached nodes if the strut is a core strut. The end-effector is moved from the rest position to a location directly above the tray slot. A logic algorithm is executed to close any receptacle fingers which may hit receptacles on adjacent struts with preattached nodes.

There are two techniques that may be used to define the location of points for a robot arm. One technique is to store an absolute location of tool tip coordinates, and the robot controller always uses the same kinematic solution for that point thus achieving an accurate repeatability. The second technique is to specify a point in terms an offset or a point that is relative to an absolute point. Although this method may not be as
precise as specifying an absolute location, it has generally provided adequate positioning.

To move to the tray slot, the location of the tray and the location of the slot are computed as an offset, or relative point, from a central taught point in the vicinity of the tray system. This approach was used because the computational storage required to keep each individual strut location would have been extensive and the time and effort to teach each of the locations would have been prohibitive. The slots are numbered from 1 to 13, beginning with the slot closest to the robot, and the trays are numbered from 1 to 9 beginning with the tray at the top of the stack. The central (reference) position is located in the center (slot 7) of the bottom tray (tray 9).

For strut acquisition the end-effector is lowered to the level of the tray containing the desired strut and the end-effector platform is extended. At this position the strut grippers on the extended platform lightly contact the grasp and alignment adapter on the strut. The arm is then moved incrementally under the control of a force-feedback loop until either a sensor on the gripper jaws indicates closure or a vertical load of 89 N (20 lbs.) is applied by the end-effector to the strut. During this operation, all moments and loads in the plane of the tray, are maintained at levels below

3.6 N (0.8 lb.) and .56 N-m (5 in.-lbs.) respectively. When either of the above two conditions are met, the gripper jaws are commanded to be latched and the platform is retracted, extracting the strut from the tray.

Since the total truss assembly was the major goal of the program, the assembly scenario had to include retrieval of each strut from the canister and transfer of the empty trays. The struts were placed in the trays as noted previously to facilitate efficient packaging; therefore, the strut removal sequence could not be sequential across a tray. Also, the packaging efficiency and interference between the end-effector and the preattached nodes required that noded struts be removed prior to an adjacent unnoded strut in the same tray. The empty trays are moved from the supply canister to the storage canister by using the end-effector to grasp adapters on the tray handles in the same manner that the struts are grasped. Although most trays were substantially filled, several strut positions in selected trays were left vacant to satisfy all assembly requirements and eliminate the need to transfer trays with struts in place.

Motion Base Moves

The carriage and turntable positions (X , Y , Theta) necessary for the installation of a particular strut are predetermined and stored in a data array in the executive program. The relative location of each motion base position with respect to an axis reference point is given in the scenario included in the appendix. The positions were established empirically to permit installation of each strut within the reach envelope of the arm and not be hampered by arm singularities. When a strut is selected for installation, the strut name is parsed by the program to determine the motion base location associated with that strut. The motion base repositioning commands are transmitted, one at a time, to the PC-based program with the command order determined by a collision avoidance routine. The PC program verifies each new position and the system status data base is updated at the completion of each move. All motion base moves are performed with the robot arm at the rest position and the end-effector located above the strut canister. This position was selected to minimize the distance that the arm protrudes toward the structure.

The carriage collision avoidance algorithm prevents the robot arm and the motion bases from colliding with any part of the structure currently on the turntable, and at the same time seeks to minimize the moves, and thus the time, required by the motion bases. Two points on the carriage are considered to be the potential collision points: the robot elbow, and the handles of the empty trays in the storage canister located on the left side of the robot. Collisions may occur during carriage Y-axis moves and turntable rotations if the carriage and robot arm are positioned too close to the turntable. Due to the truss geometry; only the core struts, or those which connect the top and bottom faces of the truss structure; are potential obstructions. A data base that includes these potential obstructions is included in the VAX executive program to facilitate checks.

The collision avoidance algorithm computes the radial distance from the center of the turntable to the point on the strut where a collision can occur, (i.e., a point at the height of the robot's elbow in its rest position) and the angle between this point's radius and the turntable zero reference. If this radius for any installed strut is greater than the radius of either of the robot or tray handle collision points in both their initial and final positions, then that strut is considered as a candidate for collision. The Y-positions of both the collision candidate and the carriage collision points are then compared to determine if interference actually occurs. For motion base moves involving both Y-carriage moves and turntable rotations, the situation is more complex. Here, the sequence of the moves must be considered and the possibility that a collision may be avoided by making the moves in a particular order must be checked. If a collision is determined to be unavoidable, then a minimum distance to move the X-carriage away from the turntable to avoid collision is calculated, and that avoidance move is commanded first.

Robot Paths and Capture Sequence

With the motion bases properly positioned, the robot arm then begins to traverse a predetermined path to transfer the strut from the canister rest position to the insertion plane where it is to be installed. Each path consists of a series of points that are executed in a sequential order. Most of these points are absolute points and the arm is moved from point to point in a sequential order without pausing between moves. This technique is used because the robot controller defines the actual robot path as straight line segments from point to point and the system has no method for specifying a continuous path. Many of the paths contain common segments because the actual location of the arm is not significant. The primary consideration is that the member does not collide with previously installed members. Near the end of the path, the speed of the arm is reduced at a location called the "approach point". At this point the strut is aligned in the insertion plane 10.16 cm (4 in.) directly in front of the strut's location in the truss. The location of the approach point is critical to final positioning, as is the point where the strut is inserted (insertion point), Therefore the approach point is defined relative to the insertion point to minimize the number of taught points.

There are three different truss/strut conditions that must be accommodated when members are inserted; therefore, the final portion of the path from the approach point to the insertion point must be tailored to the particular condition. The first condition is a direct assembly that entails placing a strut between two fixed nodes already in the structure. For this case, the end-effector moves directly to the insertion point and the receptacles on both nodes are grasped for installation. The second condition involves the installation of a core strut (one which fits between the top and bottom planes of the structure) with a preattached node at one end. Here, the end-effector also moves directly from the approach point to the insertion point, however, only the receptacle fingers (and locking mechanism) at one end of the end-effector operate. After installation, the strut is left cantilevered from the truss. To minimize the effect of gravity induced strut deformations, only the core struts were installed in a cantilever condition with a node preattached. The third condition involves the installation of a member which connects the free end of a cantilevered member with the remainder of the structure. For this condition the end-effector must move from the approach point to the deflected position of the cantilevered core strut, grasp that receptacle, then move to the insertion point before grasping the other receptacle. The intermediate (cantilever) receptacle capture points for this condition were defined relative to the insertion point and the repeatability of the deflected struts have made this a successful procedure. Obviously, for a zero gravity space condition this would not be required. To accommodate the strut cantilever deflection, as well as several other conditions influenced by gravity, the procedures developed for this demonstration test may be more challenging than would be required for space operation.

The accuracy and repeatability of the motion bases and the commercial robot are individually sufficient to perform the structural assembly. However the combination of the positioning errors of the manipulator system and the relatively high stiffness of the truss can generate significant loads when the receptacle fingers of the end-effector are attached to the joint receptacles for strut installation. To minimize the effect of these loads and assist with error free operation of the end-effector mechanisms, a wrist-mounted force/torque load cell was installed between the end-effector and the robot. All loads are measured with respect to a reference frame located midway between the centers of the receptacle fingers. The force/torque cell is nulled at the approach point for each strut. After the end-effector fingers close on the receptacle, the loads are checked to determine if they are above the predetermined threshold limits of 3.6 N for loads and 0.56 N-m for moments (typically, the loads do exceed these limits). The arm is then repositioned as necessary, using controlled incremental moves. The passive guidance features designed into the truss and the end-effector guide the repositioning to the exact insertion point. The balancing algorithm consists of calculating the displacement necessary to eliminate the largest force or moment using empirically derived stiffness gains. Also, empirically derived limits are imposed so that a calculated move that substantially exceeds the limit, and may potentially damage the mechanisms, cannot be executed. After each move the loads are reevaluated and the process is repeated until the loads are below the threshold limits. The number of cycles generally required to reduce the forces/torques below the threshold values is between 10 and 30 and the final arm displacements are typically in the range of thousandths of a centimeter. Following the arm repositioning, the strut is moved forward by the end-effector platforms inserting the joint connector into the receptacle.

The attachment of the receptacles must be coordinated with the assembly sequence. A core strut may be installed in the insertion plane in either of two directions. To maintain uniformity, the receptacles on all nodes were identically aligned and were rotationally symmetric with respect to the axis of the node. When located in the insertion plane during the approach to the node receptacles, clearance between the end-effector and the adjacent struts is approximately 1.9 cm (0.75 in.).

Approximately 19 taught paths were required to install all members instead of the 12 cell positions discussed previously. The 7 additional positions were the result of different robot base locations necessary to avoid collision situations and the orientation of the nodes in the trays (stacking requirements for the struts in the trays and the limitation of attaching nodes to only core struts).

End-effector Installation Operations

The platform is extended to insert the strut into the receptacle and the locking nut is turned by a small gear head motor, locking the strut into place. The strut latches are released, the platform retracted, and the receptacle fingers are opened to release the structure and complete the installation process. Sensors mounted on the end-effector are used to monitor the success or failure of each operation and the sequence does not proceed until the previous operation is successfully executed.

After the strut is installed the end-effector is moved back to the approach point. For those struts installed to fixed nodes the move back to the approach is direct. For core struts installed in the cantilever condition, the move back to the approach point requires that the end-effector be rolled slightly to compensate for the gravity induced deflection, otherwise the joint receptacle will frequently drag against the fingers and substantially deflect a strut before being released. From the approach point the robot moves the end-effector back along the same path to the canister rest position.

Operator Pause and Reverse

The need for a pause and reverse capability was identified early in the hardware checkout phase of the assembly tests. The ability to temporarily stop the assembly process to consider the safety or success of the ongoing operation reduced operator apprehension and established confidence in the system. The capability to pause and reverse is required for system checkout, but is also used frequently during assembly, particularly where the operation involves close clearances or a video camera must be adjusted for better coverage. The increased control accorded to the operator significantly reduces the stress associated with his monitoring task. Pause represents a nonintrusive level of intervention which, if combined with a graphics predictor simulation and display, would provide a powerful interface tool for the transfer of internal knowledge to the operator. The "hold and evaluate" feature is a key element in a supervised autonomy automation environment.

Error Recovery

Error conditions detected by sensors are reported to the executive program and thence to the operator for selection of error recovery procedures. These may involve further end-effector action, such as simply cycling the actuator, or movement of the robot arm to reposition the end-effector thus minimizing interference and permitting the component to function properly. The robot arm motions are either in the form of manually commanded adjustments in arm position, as determined from the video displays, or by the force/torque balancing algorithm. For the manually commanded position adjustments the operator is prompted to enter the displacement and rotation that the end-effector is to be moved in the end-effector coordinate reference frame. This requires the operator to estimate these distances from the visual information provided by the video cameras. Inexperienced operators may require supplemental support such as models of the truss hardware or computer generated graphics displays to aid in this task.

All error recovery actions are commanded by the operator as selections from menus which are specific to each error. The only exception to the operator intervention is a force/torque balance that is automatically invoked by receptacle finger closure errors. This was automated because these errors are easy to correct and it is not necessary to involve the operator. The error recovery menu also includes two selections which do not involve corrective actions. One is an ability to ignore the indicated error ("go on anyway") when the the operator can determine from a video camera, or based upon his experience and the nature of the error, that the indicated malfunction is due to a sensor failure or is deemed not to be serious enough to warrant action. In this case, the operator assumes responsibility and the assembly sequence proceeds as if the action was successful. Another operator option is to "give up" trying to correct the error. In this case, the system is rolled back to the beginning of the command it was currently executing and stopped. This is done so that the operator is not required to remember any operational sequences.

Tray Transfer Operations

The transfer of empty strut trays to and from the storage canister is intended to be a simple operation involving no logic other than the pause and reverse capability. The robot arm always traverses a fixed path from one canister to the other. Whether the operation involves getting or storing a tray determines the order in which the canisters are accessed. The tray pick-up with the end-effector is identical to acquiring a strut from a tray, and in fact utilizes the same end-effector routines. The tray release function, however, differs from the strut replacement in the tray, in that when the end-effector reaches the release point, the strut latches are simply opened and the arm is commanded to move down in small increments until a downward force of 156 N (35 lbf.) is being applied to the tray. This sets it properly in the spring-loaded pins which hold it in place.

The height of the tray acquisition and release points within the canisters is a function of the tray number and is calculated in increments relative to the taught point representing the bottom tray position in the respective canister. This positioning scheme is identical to that used for the struts and described in a previous section. The trays traverse up and down within the canister against nylon slides on the corner posts. The initial tray transfer operations were conducted with the trays fully supported by the strut latches on the end-effector. The tray sides would occasionally rub against the nylon slides which, because of the width of the trays, imposed a large reactive moment on the end-effector mechanism. To aid in counteracting this moment, a fork shaped bracket was designed and installed on each tray which fits on both sides of the end-effector tubular structure. The bracket also aids in aligning the trays during the transfer and stacking operation.

Truss Disassembly

To remove a damaged strut from the truss or to follow an alternative sequence due to some failure, the system must be capable of disassembling the truss. Therefore, each component of the system was designed to be capable of removing the struts from the truss structure and storing them in the canister trays. The empty trays are also retrieved from the storage canister when required and placed in the working canister to be filled. To disassemble the truss the struts are removed in precisely the opposite order from the assembly sequence. It should be noted, however, that the sequence of component operations required to remove struts is not a simple inverse of the original install sequence. For example, the cantilever conditions of a strut are not the same when it is installed as when it is removed resulting in a significant change in the operational sequence of the end-effector. Therefore, separate logic routines are invoked for the disassembly operations.

Force/torque positioning is used to guide the end-effector for the removal from the structure and insertion of the strut into the tray slot. Operator monitoring, pause and reverse operations, and error recovery are identical to the installation operations.

SOFTWARE STRUCTURE

Executive Program

The executive software program, shown schematically in figure 11 with typical commands, was designed in a top-down manner. The program structure is hierarchical to support the "supervised autonomy" mode of assembly operations. The program was developed around the operational scenarios and the requirements of the hardware actuators and sensors. Projected scenarios were reviewed and terminology to clearly differentiate the various components were developed and modified several times, however, once they were established they were strictly adhered to during code development. This resulted in a modular program that reflects the total system's architectural and functional capabilities. The software structure (figure 11) is divided into five levels: planning, truss element, device, component, and verification. Commands may be entered at any level so that the system will operate in either a completely automated mode, where high level commands are automatically decomposed into lower level commands, or in a manual mode where any level may be entered directly. Commands can only call routines which are lower in the hierarchical structure and in the same branch. The highest or planning level is designed to be coupled with an automated assembly sequence planner which is not currently implemented. The automated planner output is represented by a file of "FETCH and CONNECT" text commands. These are processed by the second level (truss element) which decomposes them into individual device commands in the following order: position the motion bases, position the robot arm, and operate the end-effector. The truss element level also offers the operator the ability to install or remove individual struts at any time.

Device level commands for the carriages, robot arm, or the end-effector address the subsystems individually and make it possible for the operator to direct each subsystem manually. The device level commands are also decomposed into sequences at the component level which actually perform the specified operation. Although commands at this level are device specific, their successful execution may require interaction with a second device whose execution involves another processor. For example, arm repositioning may be required during the "INSTALL" sequence to reduce loads on the end-effector mechanisms.

The command decomposition sequence at any level is not fixed, but varies with the type of strut being installed and any special conditions determined from the data base information for that strut. Data base checks are performed to insure that the specified strut's status is consistent with the operation to be performed; i.e. a strut to be removed must be currently installed in the structure. The database is kept current by status updates as each low level command is completed and verified by sensor checks. Unsuccessful resolution of a problem results in initiation of an inverse sequence to roll the assembly process back to a state from which replanning may be performed. The failure information is passed back up through the software hierarchy, displayed on the operator's console, and the assembly process is paused.

The operator's menu display, shown in figure 12, exhibits the same hierarchy as the executive software program. The commands currently being executed, including the decomposition of high level commands, are highlighted on each menu. This allows the operator to follow the action and observe the trace to the current status. Special windows are also displayed to identify the strut being manipulated and the status of the device-level components. Dialog windows provide a running description of the success or failure of the component currently operating and prompts the operator when a menu selection can be made. An error menu is displayed when a problem is encountered and a dialog message gives additional information on the error condition. Operational experience has demonstrated that since the hierarchical command/display structure were designed around the operational requirements of the hardware they are very successful in keeping the operator abreast of the assembly process and aiding him in dealing with the problems encountered.

The system software was developed around the hardware requirements and the defined operational scenarios using standard software engineering practices. No consideration was given to structuring the software around the NASA Standard Reference Model (NASREM) architecture described in reference 5. However, a subsequent comparison of the resulting program structure with the NASREM architecture has proved interesting. The NASREM architecture is depicted in figure 13a and the automated assembly executive software structure in figure 13b. As can be seen, the automated assembly hierarchical structure corresponds closely with the four lowest levels of NASREM. For example, the NASREM primitive level corresponds with the automated assembly device level which includes the robot arm, the end-effector, and the motion base. The NASREM element move level corresponds to strut operations (fetch, connect, remove) in the automated assembly hierarchy. Figure 13b includes only those functions at each level which were needed in the automated assembly application. Other activities, such as typical operator commands at each level and error recovery, are included for completeness. Operator command capability is provided at all levels, however in practice, it is seldom used below the element move level (fetch and connect) except for system checkout.

All hardware actions and sensor processing occurs at the component (NASREM servo) level. Also, all error conditions are resolved by either operator intervention or automated actions at the component level. Unresolved errors are passed back up the hierarchy initiating an automatic reversal of the tasks performed at each level. For the assembly task, alternative actions are available only at the planning level, which take the form of substituting struts for failed members. This requires replanning the assembly sequence. Aside from the component level, the only other testing is performed at the element move level. These tests involve physically exercising and testing the locking mechanism immediately after a strut is removed from the canister. Another test has been performed immediately after locking a strut into the structure by attempting to retract the strut platform before unlatching to verify the integrity of the joint lock. A failure of either of these tests results in the selection of a substitute strut and a replanning of the assembly sequence. The world model information base (global memory) is updated at only two levels: the device level and the element move level. At the device level, the end-effector status model is updated at the completion of each component action. At the element move level, the truss structure model and the storage canister status is updated with the installation or removal of each strut.

The NASREM architecture provides good conceptual agreement with the automated assembly application, although not all activities have an entry at every level. The hierarchical model does provide a particularly concise display for operator visualization. The hierarchical structure is capable of supporting several assembly operations by providing a standard interface between the levels. For example, standardization of device-level primitives for several end-effector configurations used for different assembly operations would allow using the same executive to direct different tasks. Device level commands could be carried out on a microprocessor mounted to the end-effector. A microprocessor-based device at the component level has the added advantage of freeing the operator from the details of the end-effector mechanisms. Analysis of the automated assembly software has demonstrated that the results are very similar to the NASREM architecture.

The executive software program was initially implemented in FORTRAN to verify the operation and structure of the specified command hierarchy. The FORTRAN prototype was used to check out and refine the assembly system operation, particularly the robot state and path sequence logic, and the error recovery procedures. When the assembly system logic and procedures became stable, work began toward developing an automated system using an expert system building tool. The results thus far indicate that considerable savings in both development time and code size are possible. The expert system provides an inference engine for rule evaluation and a framework for the information bases. The FORTRAN program will still be used to provide the menu displays and I/O interfaces to the operator, the robot processor, the carriage control software, and the end-effector program. The information base resides in a shared permanent storage on the VAX. Information included in this permanent storage is restricted to system status which must be retained from one assembly test to another and information which is shared between the FORTRAN code and the expert system.

Robot Program

The program to control the robot motion is written in an expanded version of BASIC provided by the robot manufacturer and is executed on the robot's M68000 processor. The software program structure is simple and consists of a short execution loop that calls a series of subroutines which contain the instructions to perform specific commands. Command functions to setup, initialize, configure, and move the arm are examples of some specific commands. Utility subroutines were developed to perform special tasks such as repositioning the arm to reduce forces/torques indicated by the load cell or operator directed position adjustments.

The geometric data for the 19 possible robot paths and the tray storage path segments are stored in a point definition library or PDL file on the robot processor. The data is stored according to the path names specified by the command string, so that execution of the utility routines concatenates the path name onto the robot's "MOVE" commands and the desired path is traversed. When the arm has completed execution of the specified command, a "DONE" is returned to the executive program. If a robot motion error occurs during the routine, an "ERROR" message which includes the robot system state is returned. Interruption of the robot servo power is used to pause the system during arm motions. The power shutdown is detected by the robot processor and a "PAUSED" condition is returned to the executive program. A separate monitor function in the executive program precludes sending of any robot command while the servo power is shut-off. The robot program software has been deliberately configured to represent a low-level-of-knowledge system operation. Its primary function is as a repository of the points that constitute the geometric positions for the 19 robot paths and the tray path storage segments. This is consistent with the hierarchical structure of the automated system.

System Data Structures

The software program's knowledge of system conditions resides in a shared data base which contains two basic types of information, (1) the current status of all elements of the assembly system and structure, and (2) predetermined conditions/positions which are used to direct and control the robot and motion bases. The current status information is maintained continuously to represent the physical state of the system at any point in time and thus ensure continuity of system operations. This is initialized externally and is updated automatically during test runs. The predetermined position information for the robot and motion bases are points that are associated with the installation of individual struts. The predetermined position information also describes the collision free "taught" paths by which the arm moves between the canister and the various installation positions in the truss. Figure 14 illustrates the data section which is broken down by the following elements: motion base position, strut type, robot status, tray status, current strut status, and end-effector status.

The MOTION_BASE_POSITION element stores the x, y and theta values (X_Car, Y_Car, Turntable) for the pre-determined motion base locations that establish the positioning relationship between the robot base and the truss. Due to the reach capacity of the robot, many of the struts are installed with the robot base situated at the same motion base position.

The STRUT_TYPE element contains all the data necessary to describe in detail the installation and storage conditions for each of the 102 truss strut members. The struts are identified and accessed by an alphanumeric designation (Name). The current location of the strut (Where) is accessed by the system before any strut operation can be initiated. The system must know if the strut is currently in its tray, installed in the structure, or held by the end-effector. When a strut is selected for installation, the system refers to a list of struts (Connect_To) which defines those struts that must be installed in the truss prior to installation of the selected strut. This is a safety feature that ensures the required initial conditions for installation of the selected member are satisfied. The secondary reference to the location status (Where) of each strut on this list certifies that all conditions are met. The installation position (Loc_In_Cell) identifies which of the 19 predetermined robot paths is to be followed in installing or removing a strut. If a node is preattached to a strut the record (Node_End) indicates which nut driver on the end-effector must not be operated while installing that strut. If the end-effector must capture another node, (Cap_End) specifies which end. The condition of the installed strut (Cantilever) is used to establish predefined modifications for the robot path which must occur during the capture sequence. Due to tray packing requirements, a preattached node may not be located on the correct end associated with a direct path entry. The record (Flip) identifies this condition and initiates a robot command to rotate the strut 180 degrees at an intermediate point in the installation path. To remove or insert a strut in the tray requires the assigned tray and slot position (Tray, Slot), be supplied. As noted previously, each strut is installed along 1 of 19 predefined paths. Each state in the path is defined by a robot position record (X, Y, Z, Roll, Pitch, Yaw) which are the "taught" points that are used to define the path. The collision avoidance algorithm requires the end position of the core struts be defined for computation of potential collision conditions. The record (X_End, Y_End) is used to supply this information with the strut element.

The ROBOT_STATUS element contains the path and current positioning point on the path (State, Sub_State) for all strut paths and the tray transfer path, as well as and the current strut/tray (Strut_Now, Getting_Now). This element also defines is the name of the last strut/tray installed/removed by the robot (Strut_Just_Had).

The TRAY_STATUS element maintains all information pertaining to the strut storage trays. The robot path identifier is (Tray_State), and (Tray_Mode) denotes if the objective of the move is to store or retrieve a tray. The number of the tray that struts are being removed/stored in is noted as (Current_Tray).

The CURRENT_STRUT element contains information pertinent to the status of the end-effector for the strut that is currently in the end-effector. The status variables indicate whether the nut driver sockets are seated to lock/unlock the joint connector (Left_Seat, Right_Seat) and the current status (locked or unlocked) of the joint (Left_Nut, Right_Nut).

The END_EFFECTOR element maintains the current status of the various components on the end-effector. The records for the position of the receptacle fingers at each end of the end-effector are (Left_Scar, Right_Scar) and indicate whether they are open or closed. The position of the strut insertion platform (Platform) and the condition of the strut grippers (Latch) are also maintained

Data examination is available to the operator through the main menu. This provides a direct method for accessing the status of any component and determining if initialization conditions are correct. If the operator determines that a value is incorrect or desires to manually override the indicated status he can do so, but only through password entry into the data base. This protects the system from haphazard modifications by an inexperienced operator, while permitting flexibility in control of variables for system set-up and debug. If a user modified variable affects other data items, the operator is responsible for making these changes also. For example, if the location of a strut is changed from the tray to the end-effector, the operator must change the status of the end-effector strut latches from open to latched.

End-effector Software

The end-effector software has evolved on an empirical basis. The initial task of generating actuator commands and monitoring sensor output was increased by the need to provide effective error recovery. This was developed as the end-effector mechanism error modes were identified through actual assembly tests rather than preliminary bench testing and the solutions were developed through experience. The need for a pause and reverse capability for the operator and the ability to roll back following an unresolved error complicated the end-effector sequencing algorithm and its software.

As noted earlier, the end-effector software evolved into a modular, hierarchical structure resembling the lower levels of the NASREM model. The software program to implement the end-effector command hierarchy was initially located in two places. The lower level end-effector component commands such as "LATCH" , "LOCK"' , and "CLOSE FINGER" and the actual interface to the actuator and sensor hardware were implemented on the robot processor in a dialect of BASIC. This was done because the robot processor was well equipped to communicate with external equipment and sensors. This provided an excellent testbed environment for the development of the suite of end-effector sensors, operational sequences, and error recovery procedures. The higher level end-effector component commands such as "INSTALL" , "REMOVE" , "ACQUIRE" , and "DROP" were programmed in FORTRAN on the VAX executive processor. These commands were decomposed into the lower level sequences which were then passed to the robot processor for execution. Error recovery routines and menus were programmed on the VAX in FORTRAN.

TESTS

Although the system was designed in a top-down manner; implementation has moved from the bottom up by subsystem development, testing, and integration; culminating in complete end-to-end assembly (and disassembly) of the 102 member truss structure in a supervised autonomy mode. The assembly tests conducted to date have followed a manually-developed assembly sequence based on taught paths and strut locations. The objectives of the end-to-end tests are to (1) identify through experience the type of errors likely to be encountered during assembly of space truss structures, (2) examine the reliability of the system to determine if the concepts implemented are suitable for space application, (3) evaluate the effectiveness of the operator in resolving problem situations, (4) quantify the time required for the various portions of each task and, (5) assess the operator time required to resolve the errors encountered. The testing has resulted in a baseline which can be used to estimate the effect that changes in individual operations will have on the entire assembly and a compilation of some of the errors that may be encountered along with a number of potential solutions.

Each test is directed by a command file that contains a predetermined strut sequence for either assembly or disassembly. The system executes the stream of commands as they are read from the command file. If an error occurs the system will execute preprogrammed commands to recover. If the error recovery commands fail to correct the error, then control is turned over to the operator and an error recovery menu is displayed. The operator then verifies the problem using the video system and selects the appropriate recovery action. When the error is resolved the system will resume executing commands from the command file at the step following the command where the error occurred. When resolution of the problem from the console is unsuccessful the operator's last resort is to enter the assembly area with the system disabled and manually intervene to correct the problem.

The operator uses a personal computer spread-sheet program to record elapsed times for all assembly operations, the type of errors, and the recovery technique employed. The operator's objective is to resolve the problems from the console using the menu display and video information. The spread sheet data is analyzed to identify systematic problems in the component hardware and difficult operator interactions. This is then used to refine the mechanisms and operational procedures to minimize problems that require operator attention. A complete end-to-end assembly test is not conducted in one single continuous time period since it requires 12 to 15 hours, but proceeds in four to six hour blocks so that operator fatigue is not a factor.

An assembly test is initiated with three nodes mounted to the turntable motion base and all of the struts in the trays of the supply canister. The robot is commanded to a predefined calibration position by the operator and placed into a vendor supplied self calibration routine. The reference position of all motion bases is checked to verify their calibration. After all system checks have been made and visually verified by the operator the assembly sequence is started. The timing begins with the robot arm located at a rest position near the top of the supply canister. There are seven time segments performed during the assembly of a strut. These segments are illustrated in the sketch of figure 15. The first time segment involves the movement of the arm from the rest position to the strut acquire position immediately above the strut. The second segment is the time required for the end-effector to remove the strut from the tray and involves several force/torque directed moves and many of the end-effector component operations and sensor checks. The third segment is the movement of the arm back to the rest position. The motion bases are then moved (forth time segment) to the desired location as directed by the collision avoidance algorithm. The arm is then moved along the preplanned path (fifth segment) to the approach point which is 10. 2 cm (4 in.) immediately in front of the installed location in the truss. The sixth segment is the actual install operation and involves the 10.2 cm of arm movement to the install location, force/torque repositioning, end-effector operations including locking of the joint, followed by arm motion back to the approach point. The last timing segment involves the return of the arm along the preplanned path to the rest position just above the supply canister. Several of these segments, e.g. arm motion to the install point and arm motion back to the canister rest position might appear to be redundant, however, there are differences in these individual segments that can affect the total time. If an error occurs during any segment that requires operator intervention, the time required for the operator to correct the problem, as well as the source of the problem, is recorded.

A total of four complete assembly sequences and four complete disassembly sequences have been conducted, although, only the results from two of each have been used for data analysis. The first two tests were introductory assemblies and a substantial number of changes were implemented in both the software and the hardware as a result of the initial test findings. Also the timing sequences were substantially modified following the introductory tests. The quantitative results presented herein for system timing, identification and evaluation of errors are intended to characterize the current performance of the automated assembly system.

CURRENT TEST OBSERVATIONS AND RESULTS

Preliminary Assembly Results

A summary of the results from the two assembly and disassembly tests are presented in figure 16. The results are in the form of pie charts and the sectors of the pie are the timing segments identified previously. These results are the total times for all of the 102 members and therefore, they can be used to determine representative averages for the total process as opposed to values for a particular strut. The average time for installation is slightly over 9 minutes per strut and the removal time is approximately 8.5 minutes per strut. The difference in the time required is primarily due to operations associated with removing and replacing the struts in the storage tray. The time to acquire the strut in the assembly sequence is longer because this procedure involves a force/torque balancing to move the arm into position on top of the strut closing the strut latches. This is not required for disassembly. Of particular interest is the relatively large amount of time necessary for end-effector operations during "INSTALL", "REMOVE" ,"ACQUIRE", and "DROP". These segments are relatively long because they typically involve many actuator commands and sensor checks. A number of the actuator commands are relatively slow, such as the joint locking and latch closing commands. The actuators and sensors also involve a significant amount of communication between computers which typically require about 1 sec each for the simple serial communication link currently being used. The time shown on figure 16 does not include the time required to transfer the strut trays from canister to canister. The tray transfer time required about 4.75 minutes per tray or a total of about 38 minutes per test for both assembly and disassembly.

A simple time-motion analysis of the assembly was conducted for which every operational command was identified and a estimated time to complete each command was assigned. This analysis indicated that the assembly time could be significantly improved and that it could be expected to be performed within a range of 3-5 minutes per strut as opposed to the current time of over 9 minutes per strut. For the time-motion analysis all actuation commands were considered to be driven by electrical motors that required 3-5 seconds to complete as opposed to the current pneumatic actuators which take a much shorter time. The time required for arm motion, which is associated with the robot speed, was not reduced because the operator must have adequate time to react in order to prevent collisions. The assessment of the current system operator is that monitoring the arm while operating at speeds higher than those currently being used, would be stressful and tiring. The major improvement in the assembly time occurred in end-effector operations associated with acquiring the strut from the tray and installing it in the truss. A microprocessor and associated software to command actuators and poll sensors has been developed and has been implemented in a second generation end-effector. This new capability is discussed in reference 6. Retrofitting the current end-effector to be controlled by this microprocessor could easily reduce the time for the strut installation by a factor of 2, and it has the potential for a reduction of up to 5. In developing the time-motion estimate it was also assumed that a distributed control system with parallel execution of some operations could be implemented. The main advantage of this is the elimination of the time required for motion base moves (nearly 1 min.) because they could be accomplished while the robot was moving to the tray and acquiring the strut in the end-effector.

This preliminary timing information may be representative of what is possible with an automated assembly system and it may appear relatively slow in comparison to astronaut assembly times of about 30 sec per strut reported in reference 7. However, increasing the speed of assembly operations is not the major objective of this research program. Of much greater concern is the overall reliability of the system in completing the assembly task and the ability of the system monitor to successfully correct errors when they occur. Since no EVA is planned as a part of these assembly operations and the system monitor could supervise operations from a terrestrial based control room, all that is required is that the system proceed at a rate adequate to accomplish the desired task in a reasonable (as yet undefined) time.

The number of errors that occurred during the two complete assembly and disassembly tests ranged from a high of 74 for one of the assembly tests to a low of 28 for a disassembly test. For this preliminary test set, more errors occurred during the two assembly tests than during the two disassembly tests. In all tests over 90% of the errors that occurred were the result of displacements that either prohibited the end-effector from properly functioning or caused difficulty in removing/inserting a strut in the tray. All positioning problems were corrected by the operator using either the automated force/torque balance routine or the adjust capability to slightly reposition the arm. The adjust moves are typically in the range of 5.1 mm (0.2 in.). The average time required for the operator to confirm the problem, make an independent assessment, and complete the correction was about 2 minutes per error. This is indicative of the effective error recovery procedures which have been developed. Although the timing profiles discussed previously do not include this error time it does not contribute significantly to the total assembly time. The remaining errors were the result of a variety of mostly minor problems, several of which were so inconsequential that the operator was able to determine that it was safe to proceed without taking a corrective action. Only about 1% of the struts required the operator to manually intervene and most of these were associated with errors in the length of the truss members that caused difficulty with member insertion. The truss is a redundant structure and although the length of each member is nominally the same, there are small differences in the member lengths due to fabrication errors. The joints were designed to compensate for some error in length and to have passive guidance features that would assist with misalignments. These guidance features, however, did not have adequate ramping and modifications to the basic design have been initiated.

The quantitative results for both assembly times and the errors encountered are presented more for interest and completeness than as a prediction of the achievements capable with a significantly more sophisticated space based system. The system is in an evolutionary development process, therefore, these results must be considered as a "snapshot" in time during the preliminary phase, and not the best that can be achieved with an automated system. As the evolution proceeds, the assimilation of advanced techniques (discussed in a subsequent section) such as machine vision are likely to significantly affect both the installation time as well as the number of errors that will be encountered.

Test Observation Results

Normally in a discipline oriented research program, a systematic approach is taken to implement improvements and assess the resulting changes in performance against some established reference or baseline. That approach is unsuitable for this study for several reasons. First, this program is structured around the development of a generic technology with application to a number of proposed missions, as opposed to being developed for a particular project for which a set of guidelines and requirements have been established. Second, identifying specific requirements, communicating them effectively, and assessing the impact of a proposed approach is more difficult when the research crosses discipline lines. Third, there is no data base or reference to judge progress or improvements against. Since there is no established methodology, the research is very dynamic with small perturbations occurring constantly. For example, in the process of evaluating an approach which is either unsuccessful or marginally successful, an approach that may be very successful will become evident. Therefore, the most important and potentially useful results at this stage of system development are qualitative results and observations. In addition to the specific items noted below, a major realization from this test experience is that most of the problems encountered and resolved would not have been evident using solely, graphic simulations or bench tests of separate components. The following is a compendium of the observations and results to date:

(1) Integrating robotics into the initial design has been a key element in the success of this program, as opposed to retrofitting existing hardware, developed for astronauts or other assembly methods to automated telerobotic assembly operations.

(2) The development of test hardware that is designed around the use of simple components that are commercially available has aided in the ability to initiate tests quickly and to overcome problems through hardware modifications as well as software adaptation.

(3) The concept of having the end-effector grapple the structure while inserting the strut in the receptacle has proven to be operationally successful. Using the robot arm to push the strut directly into a receptacle, which was considered early in the development, simply would not have worked. Any drag from friction that occurs during insertion, or misalignments, will move the receptacle and thus compound the problem. Also, for many truss structures that require assembly the direction of insertion cannot be aligned to lie along the axis of a supporting member to react the insertion load.

4) Reliable strut installation was achieved only through compliant moves provided by a wrist-mounted force/torque feedback system. Misalignments, (although they may be small) which occur through normal positional errors, cause high loads to be induced in the end-effector. A stiff robot arm attached through the end-effector to a stiff truss makes some form of compliance necessary to relieve the loading on the end-effector operating components. During force/torque repositioning, the final positioning of the arm typically involves motions as small as 0.002 cm.

(5) Passive guidance features designed into the end-effector and joining mechanisms have significantly aided capture of the joint receptacle by the end-effector fingers and installation operations. Modifications to the passive alignment features of the joint receptacle, as indicated earlier, are required to overcome installation difficulties.

(6) Positive actuation of all capture and release functions is mandatory. Spring actuated release mechanisms were initially incorporated into the design of the end-effector and they proved to be unreliable due to loads caused by misalignment. Also solenoid-actuated latching devices were not effective. A robust positive force capability is needed for all power driven actuation devices.

(7) Full instrumentation of the end-effector is essential, particularly for automated operations. Checking each step, using simple instrumentation sensors, made end-effector operations very reliable. Approximately 33 tests are performed by the end-effector in removing each strut from the canister and installing it into the structure.

(8) Video camera coverage is necessary for observing end-effector component operations and verifying the end-effector instrumentation.

(9) Surveillance and over-the-shoulder video camera coverage is essential for operator viewing of system operations. However, the complex paths, the lack of depth perception, and the number of camera positions required to ensure collision free paths, appears to make teleoperation for any task other than error recovery difficult. Teleoperation capabilities for complex assembly paths and tasks similar to those examined in this investigation need to be evaluated.

(10) The capture and installation of gravity-deflected cantilevered struts presented fewer problems than originally anticipated, due to the repeatability of the strut's deflected position.

(11) The assembly process must be under computer control. It is difficult for even highly experienced operators to remember system status and operational sequences. All manual commands and operator intervention should be verified by knowledge-based tools to ascertain their advisability and safety before execution.

(12) Complete database information is required on the status of the system to make decisions and resolve conflicts. The database information must be checked and updated by system sensors at program initialization and kept current by sensory input at the lowest level possible. The database update must be done automatically rather than being left up to the operator. All levels of the command hierarchy check the database prior to issuing each command to avoid unnecessary or conflicting commands.

(13) The software was designed primarily from the point of view of the operator and his role in controlling the system and handling errors. The resulting modular hierarchical format has remained virtually unchanged during the evolution of the system and it has proven to be very successful.

(14) Errors detected at the component level must be resolved before the assembly sequence can continue. The error condition is passed up through the command hierarchy and, in some cases, can be resolved by alternate actions at a higher level.

(15) The dynamic hierarchical menu displays provide a concise summary of what the automated system is doing and why, and are essential to the operator's understanding of the system's current status and its progress in the overall assembly sequence.

(16) A manual "adjust" capability to command small changes in the end-effector position is a desirable and useful feature. The adjustment range should be limited by the software to avoid collision in the event of operator entry error. A teleoperator mode may be useful and is being considered for implementation and evaluation purposes.

(17) For the operator's convenience, every operation must be interruptible, or able to be paused at any point, to give the operator the capability to survey conditions and decide if the process is proceeding in an acceptable manner. From the paused condition, every operation must be reversible to permit the operator to roll-back to a known or successful condition.

(18) Clear, concise naming conventions for the truss members, the assembly system devices, and the component commands is vital to the operator's ability to monitor and intervene in the automated operation in an expedient manner. The name designations that finally evolved are simple but the process was surprisingly difficult and many of the names were modified several times before a suitable convention was agreed upon.

(19) Bench testing of strut joining is not a substitute for "insitu" operations. The problems that surfaced during actual assembly were more complex than those that are likely to have been encountered in controlled situation bench testing. Also, tests with real hardware are time consuming and expensive, however it permits many of the real problems to surface. Graphic simulation, while necessary and effective, will not substitute for test hardware experience.

(20) In the event of an error, it is very desirable to be able to automatically roll back a composite command to the initiation point prior to handing control back to the operator. Otherwise the operator is required to recall the sequence of primitive commands at the point where intervention is necessary and he must be prepared for any error situation.

(21) Receptacle presence sensors were mounted in the receptacle fingers to detect the joint receptacle. This relieved the operator from having to monitor the capture of cantilevered struts and eliminated many of the initial error conditions.

(22) Experience has led to the incorporation of several automated error recovery features. One of these is a force/torque balance for receptacle finger closure errors so that the operator is not bothered by routine easily corrected problems.

23) The force/torque balance algorithm has been limited to 30 iterations to safeguard against limit cycling.

(24) The receptacle fingers are the most vulnerable end-effector component and several sets have been distorted due to robot position errors and loss of robot and/or motion base calibration. Since errors are likely to occur, the most vulnerable end-effector component must be capable of taking a substantial portion (or potentially all) of the full robot force capability without failure. An alternative would be to use the failure value of the most vulnerable component as a robot control limit.

Subsystem testing was helpful in the early development phase, however, it became apparent that tests of this type would not identify many of the problems encountered in the full assembly tests. Most of the problems encountered in the full assembly tests were problems which could not have been anticipated. For example, bench testing of strut installation with the end-effector would not simulate all conditions encountered in the actual in-situ installation of a strut in the truss. It became apparent quickly that few struts could be installed, even with the passive guidance that had been designed into the hardware, without the use of force/torque feedback positioning. Most end-effector operations were initially performed manually by initiating commands at the component level. This demonstrated the need for additional sensor feedback and helped to develop the scenario for the correction of errors and the need for automated roll-back to the condition that existed prior to the error. As the higher levels of automation were integrated the value of the dynamic hierarchical menu display was recognized along with the need for a "Pause" and "Reverse " capability. A significant portion of the system capability, especially with regards to software requirements, was empirically developed from the problems that were encountered, most of which could not have been anticipated without the hardware experience.

The merging of basic technology in robotics, structures, and mechanical design into an interdisciplinary effort that has used a generic building block as a research focus has been successful in advancing the technology in a relatively short time. Addressing the total integrated task, as opposed to component testing, has forced all aspects of the task to be evaluated. No problems have been encountered which would indicate that automated assembly is not a viable option for in-space construction. All of the fundamental processes have been addressed and the system is fully functional in an supervised, automated mode. This hardware testing indicates that the basic methodology developed in the areas of mechanisms design, assembly techniques, software structure, and operator interface will apply directly to a space-based automated assembly systems.

FUTURE RESEARCH OPPORTUNITIES

The research conducted to date has provided insight into additional studies that are needed to advance the current state of technology. Most of the topics listed in the subsequent sections have been initiated and the research is in differing stages of development. The intent is to identify these activities and thereby demonstrate the breadth of the research activity

Vision System

The practice of operating a structural assembly system that relies totally on the use taught points for strut capture and installation is neither flexible nor robust enough for space operations. Therefore, work on a sensor guidance system has been initiated to develop and evaluate the capability to locate and guide the robot to a passive target mounted on the joint receptacle. Another benefit of a sensor guided system is its ability to aid in the implementation of a second generation end-effector that is capable of installing struts one end at a time.

Several alternatives for locating the joint receptacle and guiding the arm have been considered and a variety of sensor options are available: infrared, laser-guided, and optical. Two of these options, infrared sensors and optical machine vision, have been evaluated. The use of infrared sensors was ruled out after initial tests because the tests indicated the need for a large servo range to accurately track distant targets and the implementation would have required the use of multiple sensors for the total guidance operation. Therefore, attention has recently focused on the potential for use of a machine vision system. Incorporation of a vision system into the assembly operation was delayed until sufficient progress had been made in the vision development and the assembly scenario requirements.

The telerobotic delivery of a strut from the canister to installation in the structure is projected to be made up of three phases: (1) - an automated path planner which uses geometric information on the structure, robot arm, and carriage positions to guide the arm to a position where, (2) - a camera mounted on the end-effector can locate targets on the node to which the strut is to be attached and guide the arm to a position where the end-effector fingers can grasp the joint receptacle. At this location the, (3) - force/torque feedback can be used to precisely align the end-effector for strut insertion. Although a path planner for the automated assembly system has not been evaluated, it is estimated that such a tool should be able to guide the arm to a point within approximately 0.46 m (18 inches) of the strut's installation point and be located within about ± 5.1 cm (2 in.) of the desired location in each of the three translational tool tip coordinates. This point is denoted as the Vision Approach Point. At this point the vision system would discriminate and locate the target, then align and guide the robot to a point near the joint receptacle. The targets must be specifically designed, accurately fabricated, and positioned on each node receptacle to provide the information necessary to guide the robot end-effector to the proper alignment position with the node receptacle at about 13 cm from the node for strut insertion/removal.

A system has been designed and preliminary target acquisition tests have been conducted in the assembly test facility. Implementing the vision system introduced a number of hardware issues. One major concern was the limited space available for the location of the camera on the end-effector and the target on the node receptacle. Also, the camera had to be aligned with the target and be nonobtrusive so as not to affect the assembly scenario or end-effector component operations. Therefore, the video images for these tests were produced by miniature Charge Coupled Device (CCD) video cameras similar to those already mounted on the end-effector. Since these miniature video cameras do not have the capability for remote focusing and iris setting, the range of operation and lighting conditions must be bounded.

The primary advantages of the vision system are the capability to acquire the desired target with little or no servoing, to provide a real-time view of the image scene to the system operator, and to process the images at a rapid rate. The current image processing hardware consists of a commercially available system configured with processor boards to perform the tasks of image capture, storing, thresholding, centroiding of regions (blobs), and graphical display. It is hosted by the executive computer. Controlling software has been developed to process images through the various processor boards and provide for operator interaction.

Figure 17 shows a photograph of the receptacle target currently used which measures approximately 1.3 cm by 2.5 cm. The light-colored dots on the flat black background are made of a retro-reflective material and are arranged in a distinctive pattern to facilitate target recognition by an image processing algorithm. The current target design satisfies a variety of requirements: it is small so as not to interfere with the fingers of the end-effector; simple and accurate; capable of being mass-produced; unique enough to be distinguished in a varying and highly-cluttered background; and provide information from which range can be determined in addition to planar positioning, so that the arm can be guided to the target in three-dimensional space.

To simplify fabrication the target is constructed in layers. The bottom layer is a commercially available retro-reflective tape which has a high reflectivity for light rays that are incident parallel to the optical axis of the video camera. The top layer is a thin sheet of flat-black anodized metal to reduce its reflectivity, thereby providing a high contrast between the target dots (blobs) and the surrounding background. This black surface acts as a mask for the bottom layer and reduces the possibility of dots blending with each other or the background in the processed image. The domino configuration of five dots is simple enough to be accurately mass-produced and its geometry makes it relatively easy to discriminate from most background clutter. Only four dots are needed to determine the target's position and orientation; using five dots introduces some redundancy and allows for the possibility that one dot may be obscured. The target is illuminated by a small light source mounted adjacent to the video camera lens as shown in figure 18.

Conditions in the laboratory have not been altered to simplify the image recognition process since the goal of the vision system is to enhance the robustness of the overall assembly system. Since the desired outcome of the image processing is to extract the correct target from its background, a combination of processing tasks which take advantage of the target dot configuration is employed. The first task is to acquire an image from the video camera and store it as a gray level digital image in a frame buffer. A technique which utilizes histogram information is used to determine the appropriate gray level threshold, between zero (black) and 255 (white). This reduces the amount of information in an image that must be processed, and tends to eliminate much of the background clutter. The next step is to identify regions of contiguous pixels, called blobs, that have nonzero gray level values. Checks are made to evaluate the circularity of the blobs. The centroid of each circular blob is defined in terms of pixel coordinates relative to the processing board. At the end of this screening process the individual centroids of each target dot are sent to a four-point quadrangle algorithm. This algorithm uses the centroidal information of four of the five target dots located in the image plane to compute the three-dimensional location and orientation of the target relative to the end-effector. This information is passed from the vision control program to the robot arm command program which determines the corrections necessary to align the end-effector with the node receptacle. This procedure is performed iteratively until the end-effector is properly aligned with the node receptacle. If during the image processing cycle the target dots evade detection, either by incorrect thresholding or centroiding, tolerances placed on the various checks are automatically adjusted until either the target is correctly identified or the system operator chooses to intervene. If a situation occurs that prevents the vision system from identifying the target the system operator may identify the target thus enabling the vision system to proceed. Additional development tests are anticipated to evaluate the vision system performance under a broad range of environmental conditions. The interface between the vision control program and the robot command program is underway.

Installation of Panels

The assembly of a bare truss that has no instruments or surface represents only a part of the task of assembling useful spacecraft. In general, the parts of such systems will be both diverse and complex. To accomplish a host of assembly operations there is a question whether this should be performed by developing a general purpose manipulator tool or have a special tool for each operational phase. To gain some experience in the application of interchangeable end-effector tools, development is underway to expand the assembly task to include the installation of reflector-type panels. An end-effector has been fabricated that is capable of acquiring a 2 meter wide hexagonal panel at three vertex points and attaching it to three nodes on the top surface of the truss. This operation will require a coordinated strut and panel assembly scenario that is outlined in reference 10. The procedure involves using the strut end-effector to assemble part of the truss, parking this end-effector and acquiring the panel end-effector to install several of the twelve panels. End-effector change operations occur several times during the integrated assembly and are accomplished using a quick-change device mounted between the robot wrist and the end-effectors.

The robot arm tasks for panel installation are relatively simple compared to strut installation. The path and procedures for the panel end-effector are identical for all panels, only the position of the motion bases are different for the 12 panels. Some of the procedures developed for strut installation have aided developing those for panel installation. The panel is removed from a canister which is adjacent to the X motion base and moved to a location that is just above the nodes on the top of the truss. Fingers on the end-effector grasp two of the truss joint receptacles to establish position and alignment . The panel is rotated around these receptacles and aligned with the third node. Actuators lower the panel onto the three support nodes simultaneously where it is latched in place. The latches are lock-bolt type units which release the panel from the end-effector at the same time it is attached to the truss. After the panel is attached to the truss the end-effector is withdrawn from the truss to acquire the next panel. The panel end-effector will be instrumented with sensors to monitor each function and the installation will be controlled by a microprocessor similar to the one developed for the second generation strut end-effector (to be discussed later). Panel assembly tests are anticipated to begin soon.

Sequence Planning

The sequence for the automated assembly tests conducted to date was developed manually and required several modifications before the final sequence was established. It was a laborious task. For larger space trusses that may involve thousands of members or for the development of repair scenarios, even trained human planners frequently fail to detect dead-end sequences until much time and effort has been wasted. The development of a scenario for replacing a member in a damaged truss or replanning a sequence because of an unexpected error cannot be done in advance because it is virtually impossible to anticipate every failure. Speed and reliability will be very important in these replanning efforts. Therefore, a cooperative program with the Jet Propulsion Lab is underway to develop an assembly sequence planner.

The technique is based on representing the set of all assembly plans by either directed graphs or by AND/OR graphs. The nodes in directed graphs correspond to states of the assembly process and the arcs in the AND/OR graphs correspond to assembly tasks associated with joining two components or subassemblies. Feasibility conditions and cost functions are the tools used to develop and gauge the quality of each solution. The algorithm for generating a sequence develops it in a backward manner. It begins with the state goal which has all of the components connected together and ends at the initial state with all of the components disconnected. Details of the operations and procedures for the assembly sequence planning can be found in reference 8.

Path Planning

The use of predetermined, taught points and paths for delivering struts into the structure for installation is simply not robust enough for a viable flight system. The complexity of the truss geometry, the limited reach of the robot arm, the total degrees-of-freedom available for positioning, and the length of the struts presents a challenge for the determination of collision free paths. By the same token, the regularity of the truss geometry and the rod shape profile of the struts makes an automated path planning tool easier to implement. Therefore an effort is anticipated to implement an on-line automated path planner that incorporates the data base system developed for the assembly operations. Several approaches to the path planner may be explored. One technique is to define the geometry of the truss and use a combination of geometry and vector algebra. A second technique is to utilize a simple potential field model, several of which are discussed in reference 9. A path planner would be expected to deliver the strut along a collision free path to a location where the machine vision system, discussed previously, could assume control at the vision approach point. The path planning tool will be evaluated with the graphics simulation (to be discussed later) prior to tests with the assembly hardware.

Assembly of a Linear Truss Beam

One desirable objective is to evaluate the feasibility and difficulty of constructing other truss configurations using the same assembly system internal knowledge. This possibility has only been recently explored and the initial indications appear promising. The truss configuration under consideration is a linear tetrahedral beam as shown in figure 19. The beam uses the same struts and nodes as the ring structure, but rather than building complete rings, struts are added to the ends of a beam formed on opposite sides of the turntable. Detailed analysis of this scenario reveals that the automated system already contains all the information required for the installation of these struts. All that is needed is to produce a different truss configuration is an appropriate change in the assembly sequence which drives the automated system. The assembly sequence for the beam has been generated, but not yet tested.

The arrangement of noded and unnoded struts in the canister is fixed by geometry and node receptacle interference considerations. The development of a planning algorithm which uses a "node or no node" request during the assembly sequence to locate and select struts from the canister is relatively simple to implement. Current plans call for actually assembling a linear tetrahedral beam to verify that the system changes are indeed minimal and to develop and test a simple strut search and selection algorithm. As a further extension of this effort, a fixture is being developed for the turntable which will enable the beam to be constructed from one end and then pushed thru a set of tracks so that each bay can be indexed forward after it is assembled. This fixture will permit the assembly of beams that have a range of member length from one to two meters.

Second Generation End-effector

A second generation end-effector that is capable of acquiring a strut at one end only and installing it into the truss has been designed and fabricated. This end-effector will enhance the versatility of the system by providing the capability to assemble contoured trusses that have members of different lengths and to install system payload modules. Implementation of this end-effector will be challenging because the end-effector will be required to capture and connect struts that are cantilevered from the truss. Operations are not anticipated to begin until the path planning and vision systems have been fully developed and demonstrated.

The second generation end-effector hardware is very similar to one end of the current end-effector. Two strut grippers are located on the strut insertion platforms and a set of tube-gripping fingers has been added inboard of the strut grippers to aid in stabilizing the strut. Actuator and sensor functions are also similar to the current end-effector. Operational sequences will necessarily be more complex because every strut will have to be inserted at one end and then captured in a cantilevered condition and installed at the second end. This will also increase the installation and removal time. However, this new end-effector will incorporate an onboard dedicated microprocessor to command actuators and verify successful operation through the interrogation of sensors. The motivation for development of an onboard microprocessor is to reduce the number of signal lines from the end-effector and to incorporate a distributed computational capability that is capable of supporting concurrent operations. An 80535 microprocessor has been procured to control the end-effector which will communicate with the executive program upon request via RS 232 serial lines. All end-effector software code and logic is implemented in "C". This includes both sequence control and sensor monitoring functions. While the microprocessor decision was driven primarily by communications considerations, an added benefit is the ease of accommodating several interchangeable end-effectors with the same executive program and command interface. This effectively makes the end-effector a stand-alone device, capable of performing standard high-level commands, monitoring its own sensors, and providing diagnostics and operator information for its error recovery. With this capability in place, an end-effector can perform standard functions such as "INSTALL" and "REMOVE" autonomously, so the executive program will not have to handle the detail of how a function is performed, only whether it was successful. These commands may be general so they refer to whatever object the particular end-effector is dealing with and thereby the executive software can be reduced significantly. The speed at which end-effector operations are performed will be increased because the processing of sensor checks will be based on interrupts. This will offset, to some extent, the increased operational time for installation and removal. The microprocessor-based end-effector control has the added potential of speeding up total system operation by making motion base moves and some end-effector operations concurrent. An example of this would be the capability to move the motion base carriages into position while the end-effector picks up a strut from the canister.

Graphics Simulation

The large amount of internal information involved in the automated assembly system, the need for clear, concise transfer of this information to the operator, and a desire to minimize operator training and experience requirements make 3-D computer-generated graphics displays extremely attractive. The ability to graphically simulate what the system is trying to accomplish will allow the operator to accurately anticipate the operation about to occur and permit a visualization of any intervention operation that may be contemplated. Also, the ability to depict mechanisms and assembly operations from any, even a physically impossible, perspective could save in the assembly system's video coverage and lighting requirements. Another benefit of graphics simulation is the capacity to develop and check out software and system algorithms in a safe, non-hardware environment where all parameters are under simulation control. This would speed-up software development and thereby significantly reduce the cost. A final graphics simulation feature is the ability for rapid prototyping of system concepts and configurations.

A graphics simulation and display system has recently been procured and the development of a simulation is underway. One aspect that will be evaluated in initial simulation studies is the need for real-time simulation operations. It is not presently clear that the operator displays are required to operate at real-time rates or if sufficient information can be obtained, and a simulation can be effective that operates at other than a real-time rate.

Operator Training and Human Factors Evaluation

The control system was designed to perform the truss assembly with minimal operator intervention, however, there are many instances (particularly in dealing with system errors) when the computer software must present information to the operator and elicit a response. When the software was initially developed some consideration was given to the selection of names for commands that would be identified with hardware function and error names that elicit the appropriate response. No formal evaluation of these terms was conducted and personnel skilled in human factors were not consulted. In light of the current state of the system and the opportunity for using it as a test-bed for other studies, it became apparent that unskilled operator training and issues associated with human factors should be formally considered. Issues that may be evaluated are the type of information and the kind of visual aids which will be most helpful to the operator. One operator aid currently being developed is computer controlled video coverage (camera selection and pan/ tilt/ zoom) which will ease the task of monitoring operations.

To examine these issues it is anticipated that personnel skilled in human factors studies will be consulted to evaluate the current system and recommend an assessment plan. It is anticipated that this will involve some testing and evaluation using unskilled test subjects as assembly system operators. These operators will be provided with a common level of training and instruction on the hardware and software menus. This may be in the form of a video supported by hands-on-training at the console with a skilled instructor. For the actual tests, the assembly tasks will be seeded with a planned sequence of errors so that all operators will be required to deal with the same problems under the same set of conditions to accurately assess their performance.

CONCLUDING REMARKS

Techniques for construction and assembly in space other than those that traditionally rely on astronaut Extra-Vehicular Activity (EVA) must be explored. One alternative that is being examined at the NASA Langley Research Center is the use of an automated telerobotic system that uses an Earth based, or a space based, executive monitor to track the operation of robotic truss assembly and intervene only when the automated system encounters a problem and requires assistance. To examine the operations and obtain some practical experience a hardware test bed has been developed. The test bed was designed to incorporate off the shelf hardware and existing components to quickly obtain some practical operational experience. Several end-to-end assembly and disassembly tests have been conducted to date. The success of the research conducted in the test bed provides a very encouraging outlook for the realization of a space based operational system in a orbital, lunar, or planetary environment. One key item is the basic experimental approach which was developed around the detailed study of a focused problem. This proved to be essential in identifying the real issues and directing the effort toward well defined requirements rather general concepts. Few of the significant problems and test observations for the automated assembly process were anticipated prior to this effort and many initial concerns never materialized. The viability and performance projections of automated system concepts have little relevance unless they are based on detailed consideration of the task or on practical experience.

One key element is that the program was developed around a cooperative, interdisciplinary approach to the designs of both the assembly system and the structure being assembled. All aspects of the automated operation were developed from a total system viewpoint. Experience has shown that the impact of automation on the design of vehicles and systems is just as important as the functional design of the system itself and close coordination between the required functional requirements and the automation aspects must be maintained. An important factor also is the development of a system software and program structure based upon the needs and the role of the operator in error resolution. This led to effective error resolution by the operator and contributed to the current success of this supervised autonomy mode.

The test experience has highlighted the necessity for "in situ" development and demonstration in a realistic testbed of all aspects of the required system, as opposed to simple bench testing of individual components and end-effector mechanisms. The current LaRC automated structures assembly facility provides both a baseline assembly system and modification potential to accommodate this need in many technology areas encompassing both automation and structural design.

It is recognized that much remains to be accomplished before such a system will be ready for flight test operations. For the current program, however, it was decided to delay development of advanced operations anticipated to be required for a flight system until a practical experience base could be developed that would provide a clear definition for advanced research tasks. The experience gained to date has now provided that direction. The reliance on the use of taught points is not adequate for space operation. Therefore, a preliminary machine vision system that is capable of identifying and locating a passive target to guide the robot to the strut installation position has been developed. The incorporation of other tasks in the assembly operation, such as, the installation of reflector panels has been initiated. No viable automated system for space operation could successfully operate without both a knowledge based sequence planner to deal with unanticipated failures and a path planner. The testbed experience has provided a better definition of the requirements for these efforts, an excellent facility in which to evaluate their operation, and a reference data base against which they may be evaluated.

REFERENCES

1. Anon: Report of the Advisory Committee on the Future of the U. S. Space Program. Superintendent of Documents, U. S. Printing Office, December 1990.

2. Fisher, William F.; and Price, Charles R.:Space Station Freedom External Maintenance Task Team, Final Report. July 1990.

3. Rhodes, Marvin D.; Will, Ralph W.; and Wise, Marion A.: A Telerobotic System for Automated Assembly of Large Space Structures. NASA TM 101518, March 1989.

4. Wu, K. Chauncey; Adams, Richard R.; and Rhodes, Marvin D.: Analytical and Photogrammetric Characterization of a Planar Tetrahedral Truss. NASA TM 4231, December 1990.

5. Albus, James S.; McCain, Harry G.; and Lumia, Ronald: NASA/NSB Standard Reference Model for Telerobot Control System Architecture (NASREM), National Bureau of Standards Technical Note 1235, July 1987.

6. Doggett, William R.; Rhodes, Marvin D.; Wise, Marion A. ; and Armstead, Maurice F.: A Smart End-Effector for Assembly of Space truss Structures. Presented at the Space Operations, Applications and Research Conference, July 1991 Houston, TX.

7. Heard, Walter L. Jr.; Bush, Harold G.; and Watson, Judith J.: Space Truss Construction Studies, ASCE Space 88 Engineering, Construction, and Operations in Space Conference, Albuquerque, NM, August 1988.

8. L. S. Homen de Mello; and Desai, R. S.: Assembly Planning for Large Truss Structures in Space. Proceedings of The IEEE International Conference on Systems Engineering, Aug. 9-11, 1990 Pittsburgh, PA.

9. Munger, Rolf: Path Planning for Assembly of Strut-Based Structures. Published by The Center for Intelligent Robotic Systems for Space Exploration (CIRSSE), Rensselaer Polytechnic Institute, Troy, NY, CIRSSE Report # 91, May 1991.

10. Kenner, Scott W.; Rhodes, Marvin D.; and Fichter, W. B.: Component Count and Preliminary Assembly Considerations for Large Space Truss Structures. NASA TM 102604, February 1990.

APPENDIX A

Initial Conditions and Guidelines for Development of Truss Assembly Scenario

The following rules and guidelines have been formulated to assist in the development of the tetrahedral truss assembly sequences and scenarios. They address the considerations for structural stiffness and integrity during assembly, strut and node packing in the canister trays, and the robot and motion base constraints. The intent is to formalize these rules into a knowledge base for a goal-directed, automatic task sequence planner. The rules have been categorized into those applicable to the structural configuration, the tray packing conditions, and the robot and motion base constraints. By modularizing the knowledge base the assembly sequences may be easier to change.

Initial Conditions

1. The truss has 102 members and 31 nodes and (19 nodes in the top plane and 12 nodes in the base plane).

2. The three base support nodes are set in place prior to initiation of assembly. These base nodes are fixed and are used as anchor points for stabilizing the truss system.

3. The ring assembly starts by building the center tetrahedron, then proceeds to the first ring. This is followed by assembling the lower part of the second ring and the unit is completed by constructing the upper part of the second ring. This process was adopted to aid in tracking the process and avoid leaving a member out of the truss as opposed to a requirement. However, it does aid in reducing the amount of structure that may be assembled asymmetrically thus reducing the time/distance required to move the X carriage so it will clear the robot when repositioning the turntable.

Structural Configuration Considerations

4. A truss member can only be inserted in the plane of the square base of the pentahedron.

5. The entrance receptacles of the core struts are rotationally aligned.

6. A tetrahedral unit is completed before additional members are added to start a second tetrahedron supported by the first.

7. Two tetrahedrons may be partially developed and connected if they are supported by a third unit that has been previously completed. (i.e. A pentahedron may be completed before the two adjacent tetrahedrons are completed.)

8. Assembly is developed in rings and the rings are broken down into repeating subelements. Each subelement is completed before beginning the next subelement. After all subelements in a ring are assembled they are interconnected to complete the ring.

9. Only core struts are cantilevered to minimize deformations due to gravity and these cantilevered struts each have a preattached node. A cantilevered strut must be anchored to a completed tetrahedron or to one of the three fixed base nodes.

10. After installation of a cantilevered core strut, a face strut was always installed that would tie the cantilevered end to a fixed node. One cantilevered member cannot be cantilevered from another.

Tray Packing Conditions

11. Nodes are preattached to the members in the canister and are not free to be moved from one strut to another. In fact, the end-effector is incapable of removing a node and placing it on another strut . However, a prenoded strut could be removed and replaced by simply unlocking the joints if it has at least one other core strut attached when the strut is removed.

12. Each canister tray holds up to 13 struts.

13. Trays must be removed from the supply rack and installed in the storage rack in the order (and orientation) that they are stacked as there is no provision for temporary storage. The members are required to be oriented in the proper direction, i.e. the node on the correct end, when they are inserted in the truss. This is handled in the assembly sequence as a "flip".

14. For the current assembly scenario all members of a tray are removed before a tray is stored in the tray holder, however, not all positions in the trays are filled. Some spare struts could be permitted to remain in a tray while it is moved. They should be kept close to the center of the tray to minimize the inertial unbalance of the tray. The center storage position of the struts must be open to permit the end-effector to access the arms that are used to pick up the tray.

15. When the end-effector fetches struts from the tray it cannot fetch an unnoded strut immediately adjacent to a noded strut with the noded strut in the tray because of interference between the scar gripper and the adjacent node.

Robot and Motion Base Constraints

16. The robot reach and the axis constraints must not be violated.

17. The number of predefined robot paths and taught points should be minimized.

18. It is desirable to insert as many struts as possible at a given carriage and turntable location within the range of the robot arm. The robot base positions (X and Y carriage position) where chosen to maximize the number of struts inserted at a given location.

19. The turntable is capable of ±3 revolutions from its starting position. To avoid hitting the limit the rotation of the turntable was monitored and it was reversed regularly to avoid even approaching the limits.

20. The X carriage is always aligned after rotation with the 10/2 or 8/4 position of the clock.

21. The Y carriage travel limit is ±116 inches from the turntable rotation axis.

APPENDIX B

Assembly Sequence for a Tetrahedral Truss

The following table lists the manually developed assembly sequence for the two-ring, 102-member tetrahedral truss structure. This table is used by the operator for reference during the assembly/disassembly operations and contains information on strut names, robot arm paths, strut cantilever conditions, motion base positions, and tray assignment positions for each strut in the sequence. The purpose of this appendix is to describe the strut data and define the host of assembly operations/conditions that must be accounted for.

Figure A1 shows a diagram of the tetrahedral truss with the cells labelled and the strut sequence numbers indicated. All information in this table refers to figure A1 except for the motion base positions (X,Y,Theta), which refer to the facility layout diagram in figure A2.

The truss is assembled in basic sections, the center pyramid followed by the first ring. The second ring is assembled as two units, the lower section first followed by the upper outer portion. The outer portion of the second ring has member positions similar to the first ring. The following items define the notation on the sequence chart:

1) Strut designation is referenced to clock position with "12" being directly in "front" of the robot and moving clockwise around a circle looking down on the assembly, as shown in figure 10.

2) There are two viewing designations noted- the operator designation and the robot designation. The operator designation is referenced to a scheme to locate a strut anywhere in the truss by the use of a ring, cell and position location that is unique for that strut. The operator names are referenced to the clock designation at the bottom of the figure. The robot designation is referenced to the path and position the arm must assume to successfully insert the strut. The robot name is referenced to the turntable angle because this is the perspective from the position of the robot. The robot path is a function of the motion base location.

3) FREE, CAP1, CAP2, CLOSE and FIXED indicate strut capture/ release conditions.
FREE-attachment of a prenoded core strut with release of the member as a cantilever beam supported at root.
CAP1-capture of a FREE cantilevered strut receptacle to attach the present strut to.
CAP2-capture a strut receptacle in the structure at one end, move to and capture the strut receptacle at other, then move to installation point (Condition for 3 struts in first ring only).
FIXED-present member is attached to receptacles that are fixed in the structure due to the attachment of several members at both ends.
CLOSE-similar to CAP2 condition (two cantilever struts) with capture at only one end-the completion of this installation establishes the FIXED condition noted above.

4) The location and orientation for the core struts with preattached nodes is important. The notation UP (e. g. 12-2/UP) means that the node lies in the top surface of the truss.whereas DN indicates that the node lies in the bottom surface of the truss. The trailing R or F indicates that the taught position is REAL (R-the taught path moves the node from the canister to the correct face of the truss) or FLIPPED (F-requires a 180 deg rotation of the strut to place the node in the correct face of he truss)

5) The motion base positions X/Y/Theta in this sequence are the reference positions from which the robot arm paths are taught. Note that there is a fixed point of reference in one orientation only. In this orientation the 10_2 strut is always a bottom surface strut and the 8_4 strut is always a top surface strut.

6) The R or L under the Tray/Slot/Node column indicates that the node is on the right or left end of the tray and is picked up on that side of the end-effector. This is associated with the REAL and Flip notation. X NODE means that the location of that tray slot is one that normally has a node attached, however a node is not required for the strut that is currently allocated to this slot. All struts with nodes preattached are either (ROBOT DSN) 12_2, 10_8, or 6_4. There are no 12_10, 8_6, or 4_2 struts (with or without preattached nodes) due to the orientation of the receptacles.

SEQUENCE NUMBER	OPERATOR NAME	ROBOT NAME	X/Y/Θ (fig. A2)	TRAY/SLOT NODE/END
Center pyramid				
1	R1C1/10_2	10_2/FIXED	-205.647/-17.50/0.0	1/3
2	R1C1/12_2	12_2/FREE/UP/R		1/5/L
3	R1C3/6_10	10_2/FIXED	-205.647/-17.50/120.0	1/4
4	R1C3/8_10	12_2/CAP1		1/6
5	R1C2/6_2	10_2/FIXED	-205.647/-17.50/-120.0	1/7
6	R1C2/6_4	12_2/CLOSE		1/11
Center pyramid completed.				
7	R2C7/6_8	F10_8/FREE/UP/F	-197.647/-17.50/-120.0	1/9/L
8	R1C2/12_4	12_8/CAP1		1/8
9	R2C7/6_4	12_2/FREE/UP/R	-205.647/-96.18/-120.0	1/13/L
10	R2C7/8_4	12_4/CAP2	-197.647/-96.18/-120.0	1/10
11	R1C3/12_8	12_4/CLOSE	-197.647/-17.50/120.0	1/12
12	R1C3/6_4	F10_8/FREE/UP/F	-197.647/-17.50/120.0	1/1/L
13	R1C3/8_4	12_8/CAP1		1/2
14	R2C1/10_8	F12_2/FREE/UP/F	-205.647/-96.18/120.0	2/5/R
15	R2C1/12_8	12_4/CAP2	-197.647/-96.18/120.0	2/4
16	R1C1/12_4	12_4/CLOSE	-197.647/-17.50/0.0	2/6
17	R1C1/10_8	10_8/FREE/UP/R	-197.647/-17.50/0.0	2/9/R
18	R1C1/12_8	12_8/CAP1		2/7
19	R2C4/12_2	F12_2/FREE/UP/F	-205.647/-96.18/0.0	2/13/R
20	R2C4/12_4	12_4/CAP2	-197.647/-96.18/0.0	2/8
21	R1C2/8_4	12_4/CLOSE	-197.647/-17.50/-120.0	2/11
22	R1C2/12_8	8_4/FIXED	-172.0745/-17.50/-120.0	2/12
23	R1C1/8_4	8_4/FIXED	-172.0745/-17.50/0.0	2/10
24	R1C3/12_4	8_4/FIXED	-172.0745/-17.50/120.0	2/3
First ring completed.				
25	R1C2/10_8	B6_4/FREE/DN/F	-200.88/-38.0/-120.0	2/1/R
26	R1C2/6_10	6_2/CAP1	-198.88/-38.0/-120.0	2/2
27	R1C2/10_2	6_10/CLOSE		3/1
28	R2C6/6_4	12_2/FIXED	-137.454/-56.87/-120.0	3/5
29	R2C7/10_8	D6_4/FREE/DN/R	-198.88/-116.74/-120.0	3/3/L

SEQUENCE NUMBER	OPERATOR NAME	ROBOT NAME	X/Y/Θ (fig. A2)	TRAY/SLOT NODE/END
30	R2C7/6_10	6_2/CAP1		3/2
31	R2C6/6_2	10_2/CLOSE	-137.454/-56.87/-120.0	3/4
32	R2C8/10_8	B12_2/FREE/DN/F	-137.454/21.87/120.0	3/7/L
33	R2C7/6_2	V6_2/CAP1	-141.0/-19.50/-180.0	3/6
34	R2C7/10_2	V10_2/CLOSE	-133.0/-17.50/-180.0	3/8
35	R1C3/12_2	D6_4/FREE/DN/R	-198.88/-38.0/120.0	3/11/L
36	R1C3/10_2	6_2/CAP1		3/9
37	R1C3/6_2	6_10/CLOSE		3/10
38	R2C9/10_8	12_2/FIXED	-137.454/-56.87/120.0	3/12
39	R2C1/12_2	B6_4/FREE/DN/F	-200.88/-116.74/120.0	4/3/R
40	R2C1/10_2	6_2/CAP1	-198.88/-116.74/120.0	4/1
41	R2C9/6_10	10_2/CLOSE	-137.454/-56.87/120.0	4/2
42	R2C2/12_2	D12_2/FREE/DN/R	-137.454/21.87/0.0	4/7/R
43	R2C1/6_10	V6_2/CAP1	-141.0/-19.50/60.0	4/4
44	R2C1/6_2	V10_2/CLOSE	-133.0/-17.50/60.0	4/5
45	R1C1/6_4	B6_4/FREE/DN/F	-200.88/-38.0/0.0	4/11/R
46	R1C1/6_2	6_2/CAP1	-198.88/-38.0/0.0	4/6
47	R1C1/6_10	6_10/CLOSE		4/8
48	R2C3/12_2	12_2/FIXED	-137.454/-56.87/0.0	4/9
49	R2C4/6_4	D6_4/FREE/DN/R	-198.88/-116.74/0.0	5/5/L
50	R2C4/2_6	6_2/CAP1		5/4
51	R2C3/10_2	10_2/CLOSE	-137.454/-56.87/0.0	5/3
52	R2C5/6_4	B12_2/FREE/DN/F	-137.454/21.87/-120.0	5/9/L
53	R2C4/10_2	V6_2/CAP1	-141.0/-19.5/-60.0	5/8
54	R2C4/10_6	V10_2/CLOSE	-133/-17.5/-60.0	5/7
55	R2C5/2_6	10_2/FIXED	-137.455/21.87/-120.0	5/6
56	R2C8/10_6	10_2/FIXED	-137.455/21.87/120.0	5/10
57	R2C2/10_2	10_2/FIXED	-137.455/21.87/0.0	5/11

This completes the lower part of the second ring.

CW 360 TO STRAIGHTEN WIRES

58	R2C4/10_8	V12_2/FREE/UP/R	-153.79/-53.34/-30.0	5/13/L
59	R2C4/12_8	V12_4/CAP1		5/12

SEQUENCE NUMBER	OPERATOR NAME	ROBOT NAME	X/Y/Θ (fig. A2)	TRAY/SLOT NODE/END
60	R2C5/8_6	F10_8/FREE/UP/F	-129.455/100.68/-120.0	5/1/L
61	R3C7/12_4	V8_4/CAP2	-88.5/-55.555/-60	5/2
62	R2C5/8_4	12_4/CLOSE	-129.455/21.87/-120.0	6/3
63	R2C7/12_2	FI12_2/FREE/UP/F	-153.79/-53.34/-150.0	6/1/R
64	R2C7/12_4	V12_4/CAP1		6/2
65	R2C8/12_10	10_8/FREE/UP	-129.455/100.68/120.0	6/5/R
66	R3C12/8_4	V8_4/CAP2	-88.5/-55.555/180.0	6/4
67	R2C8/12_8	12_4/CLOSE	-129.455/21.87/120.0	6/6
68	R2C7/12_8	8_4/FIXED	-172.0745/-96.18/-120.0	6/7
69	R2C7/12_10	V12_10/FIXED	-90/14.87/-180.0	6/11
70	R2C6/12_2	10_8/FREE/UP/R	-129.455/-56.87/-120.0	6/9/R
71	R2C6/12_4	12_8/CAP1		6/8
72	R3C11/8_4	V8_4/CLOSE	-88.5/23.185/-180.0	6/10
73	R2C5/12_2	10_8/FREE/UP/R	-129.455/21.87/-120.0	6/13/R
74	R2C5/12_4	12_8/CAP1		6/12
75	R2C6/8_4	12_4/CLOSE	-129.455/-56.87/-120.0	7/1
76	R2C6/12_8	8_4/FIXED	-103.8837/-56.87/-120.0	7/5
77	R2C5/12_8	8_4/FIXED	-103.8837/+21.87/-120.0	7/9
78	R2C1/6_4	V12_2/FREE/UP/R	-153.79/-53.34/90.0	7/7/L
79	R2C1/8_4	V12_4/CAP1		7/8
80	R2C2/4_2	F10_8/FREE/UP/F	-129.455/100.68/0.0	7/11/L
81	R3C2/12_8	V8_4/CAP2	-88.5/-55.55/60.0	7/12
82	R2C2/12_4	12_4/CLOSE	-129.455/21.87/0.0	7/13
83	R2C4/8_4	8_4/FIXED	-172.0745/-96.18/0.0	7/10
84	R2C4/8_6	V12_10/FIXED	-90/14.87/-60.0	7/6
85	R2C3/10_8	F10_8/FREE/UP/F	-129.455/-56.87/0.0	7/3/L
86	R2C3/12_8	12_8/CAP1		7/2
87	R3C6/12_4	V8_4/CLOSE	-88.5/23.185/-60.0	7/4
88	R2C2/10_8	F10_8/FREE/UP/R	-129.455/21.87/0.0	8/3/R
89	R2C2/12_8	12_8/CAP1		8/2
90	R2C3/12_4	12_4/CLOSE	-129.455/-56.87/0.0	8/1
91	R2C3/8_4	8_4/FIXED	-103.8837/-56.87/0.0	8/4

SEQUENCE NUMBER	OPERATOR NAME	ROBOT NAME	X/Y/Θ (fig. A2)	TRAY/SLOT NODE/END
92	R2C2/8_4	8_4/FIXED	-103.8837/21.87/0.0	8/5
93	R2C1/12_4	8_4/FIXED	-172.0745/-96.18/120.0	8/9
94	R2C1/4_2	V12_10/FIXED	-90/14.87/60.0	8/13
95	R2C9/6_4	10_8/FREE/UP/R	-129.455/-56.87/120.0	8/11/R
96	R2C9/8_4	12_8/CAP1		8/10
97	R3C1/12_8	V8_4/CLOSE	-88.5/23.185/60.0	8/12
98	R2C8/6_4	10_8/FREE/UP/R	-129.455/21.87/120.0	8/7/R
99	R2C8/8_4	12_8/CAP1		8/6
100	R2C9/12_8	12_4/CLOSE	-129.455/-56.87/120.0	8/8
101	R2C9/12_4	8_4/FIXED	-103.8837/-56.87/120.0	9/3
102	R2C8/12_4	8_4/FIXED	-103.8837/21.87/120.0	9/7

This completes the assembly sequence.

Empty Slots:

tray/slots	
3	/13
4	/10,12,13
9	/1,2,4,5,6,8,9,10,11,12,13

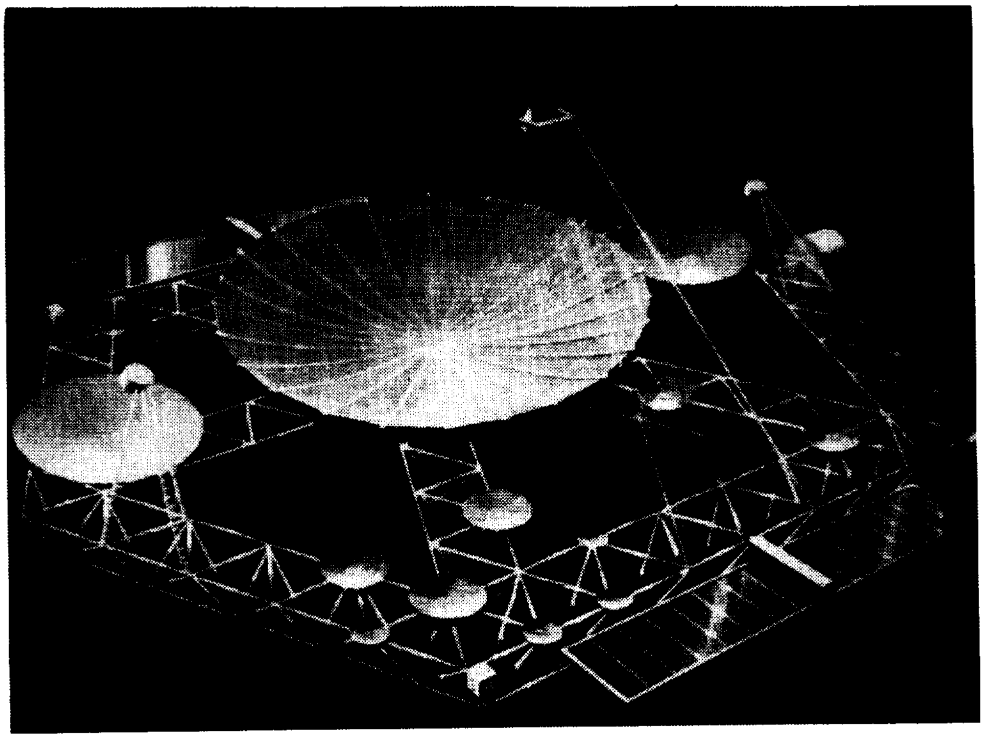


Figure 1(a). Space-based Antenna.

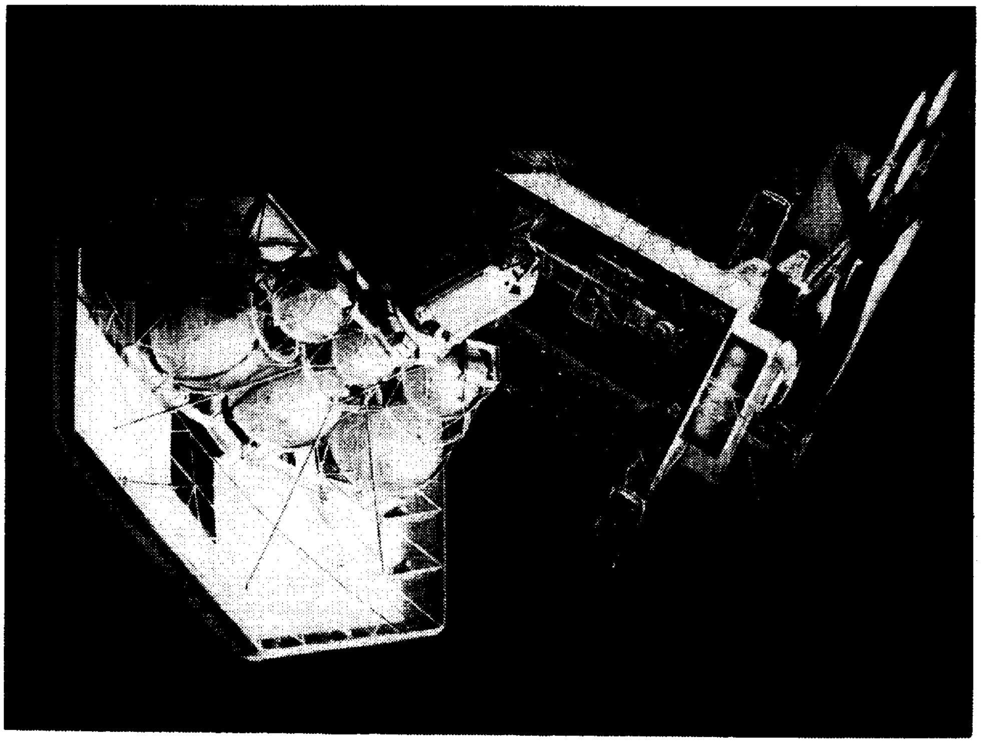


Figure 1(b). Truss-supported Aerobrake Configuration.

Figure 1. Artist's conception of future Space Missions.

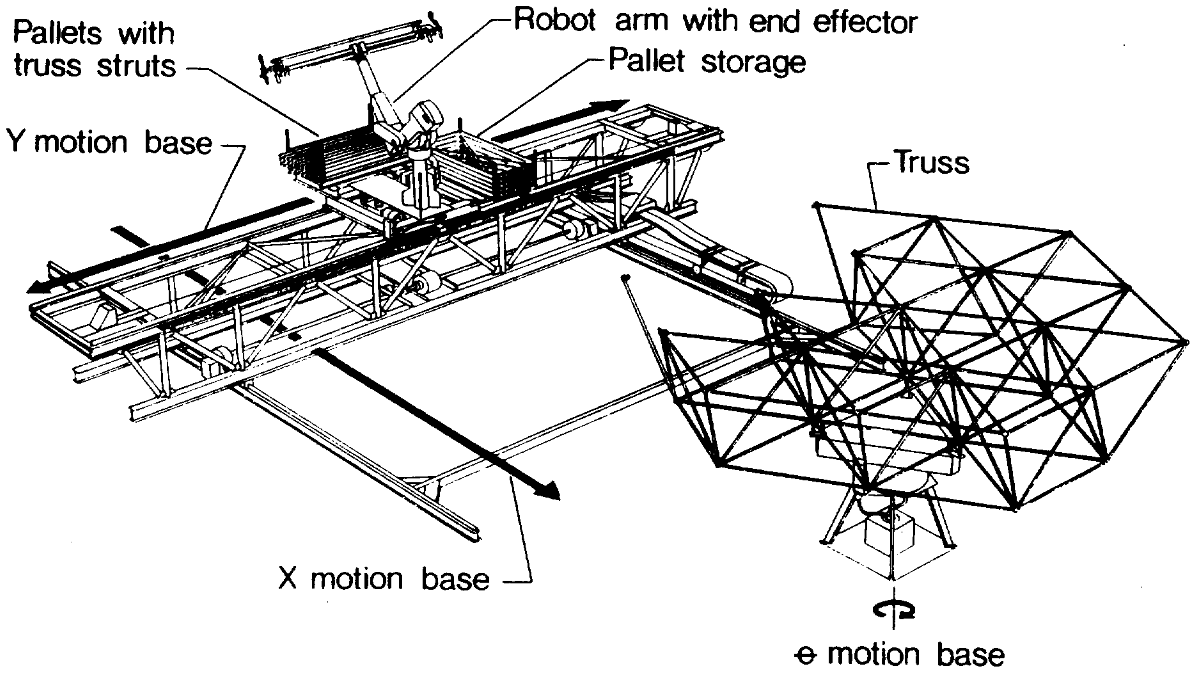


Figure 2(a). Schematic of Automated Structures Assembly Laboratory.

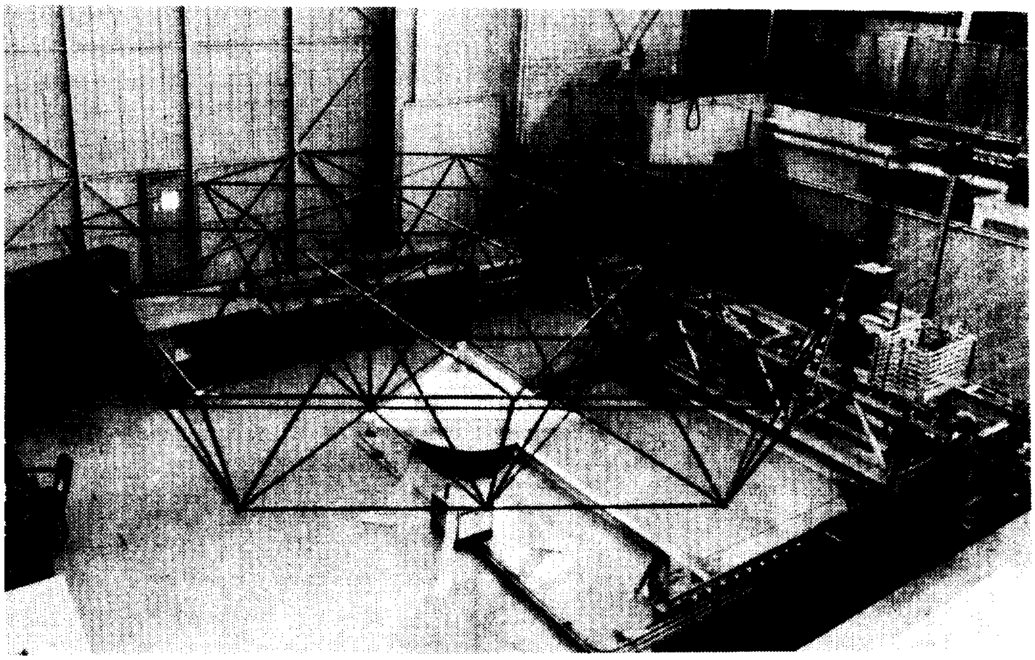


Figure 2(b). Photograph of Automated Structures Assembly Laboratory.

Figure 2. Automated Assembly facility.

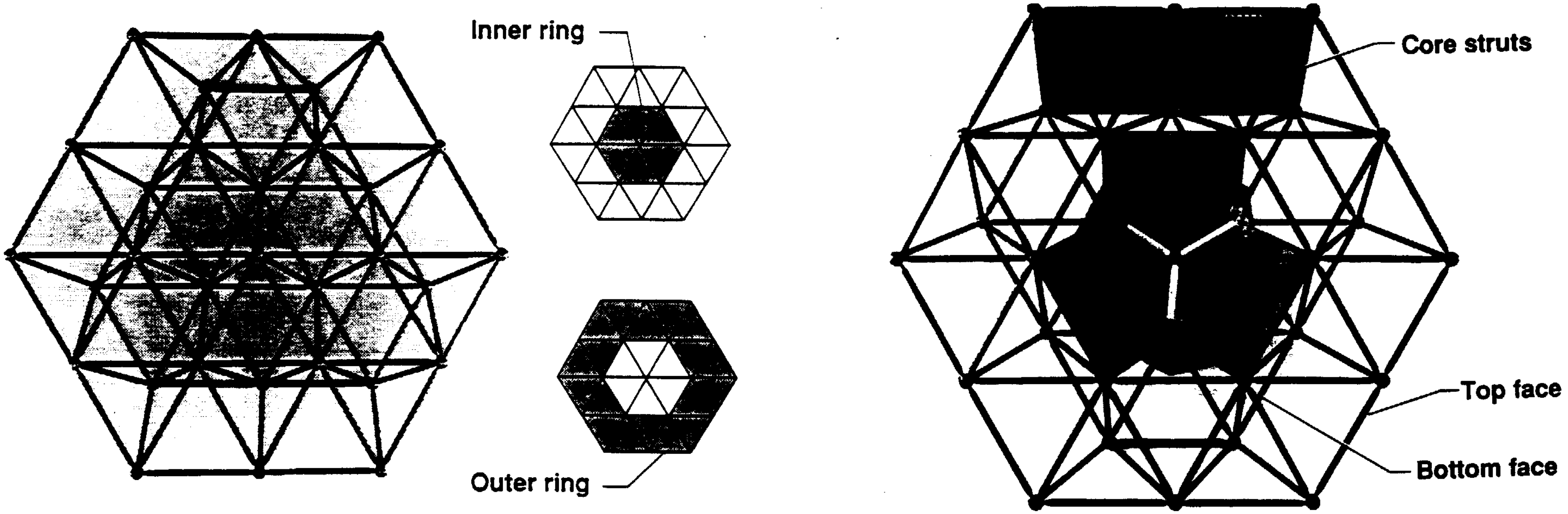


Figure 3(a). Top View of Two-ring Tetrahedral Truss Structure.

Figure 3(b). Tetrahedral Truss Structure Showing Strut Insertion Planes.

Figure 3. Tetrahedral truss structure configuration.

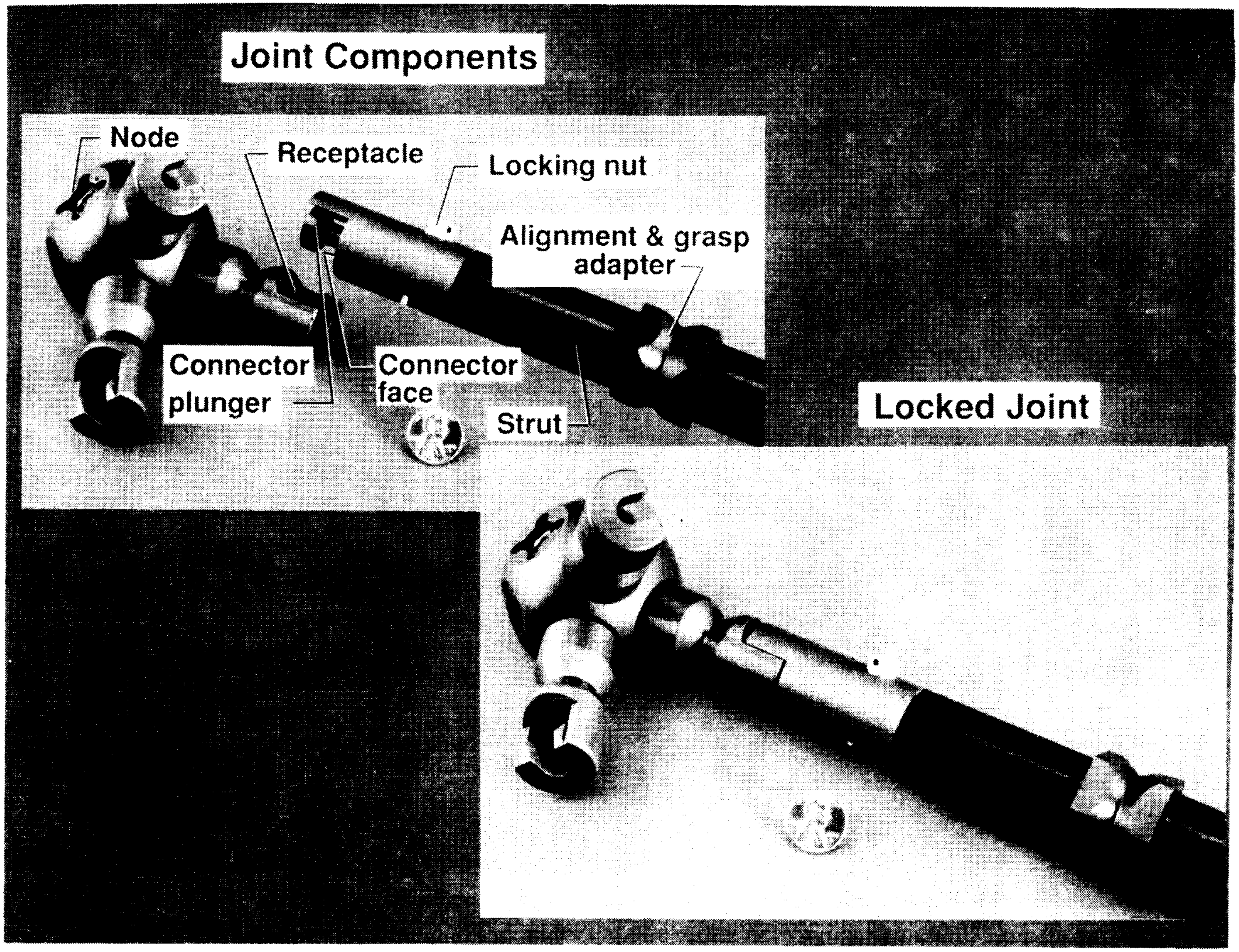


Figure 4. Strut/Node Joint Connector Hardware.

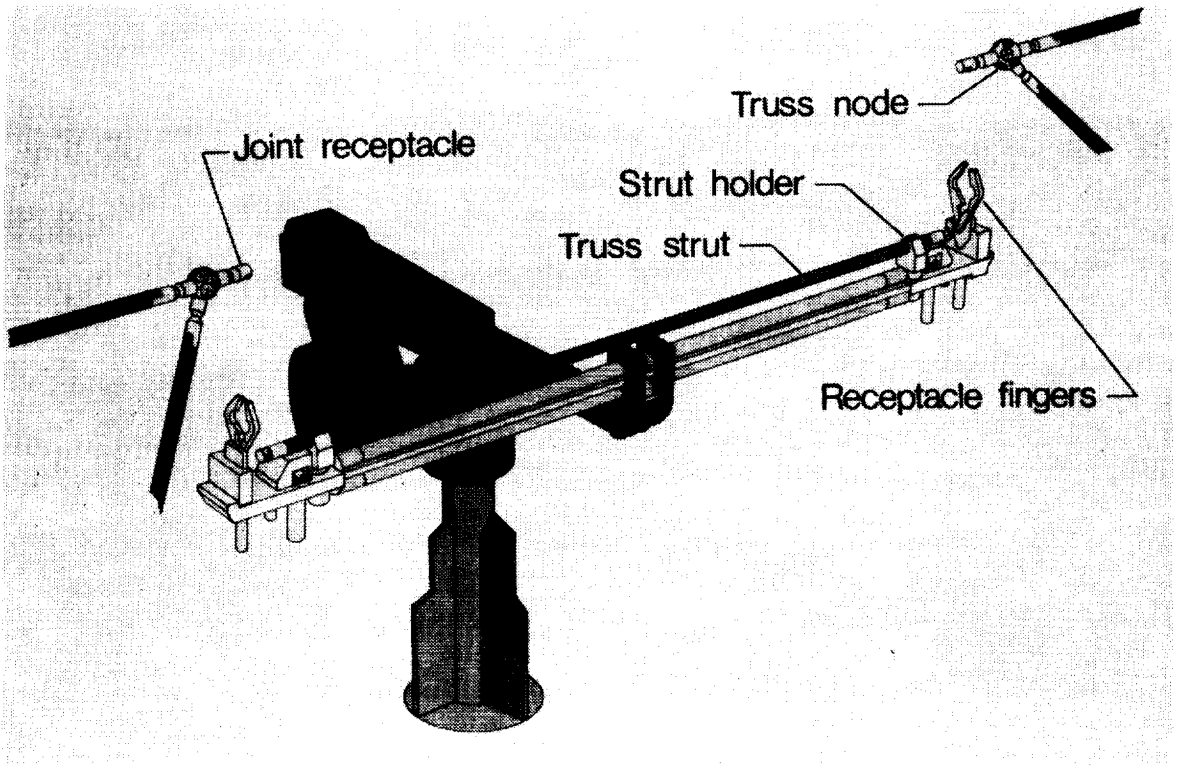


Figure 5(a). Schematic of End Effector and Strut Insertion Concept.

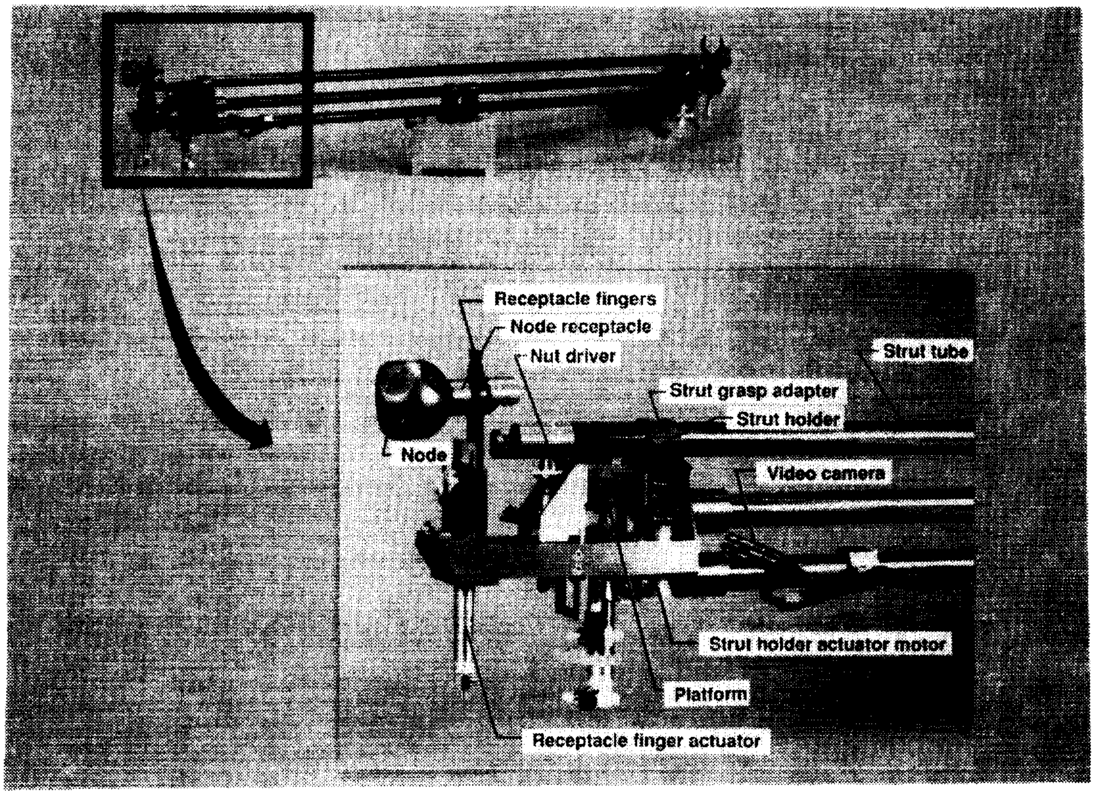


Figure 5(b). Photograph of End Effector and Actuator Mechanisms.

Figure 5. End effector tool

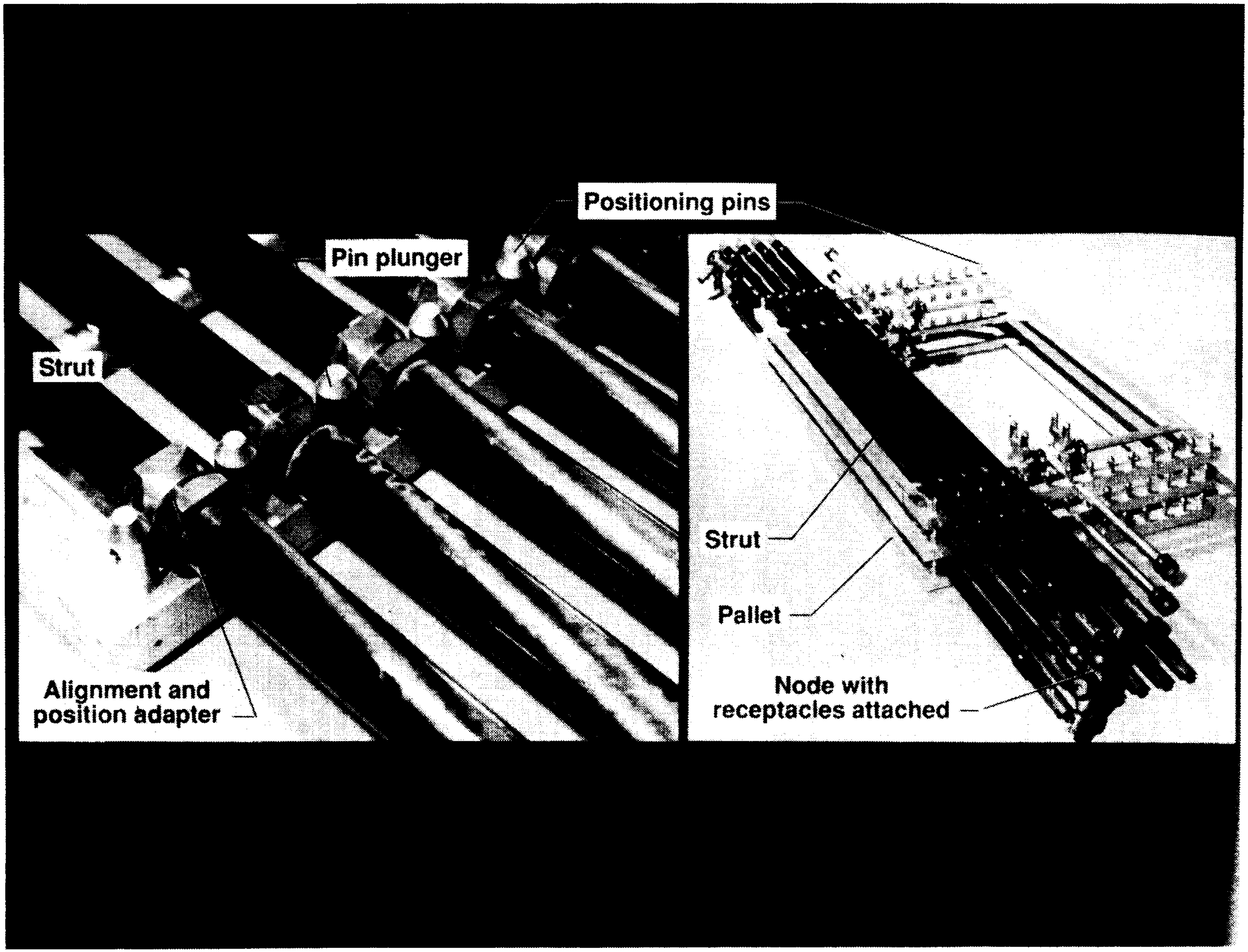

Positioning pins
Pin plunger
Strut
Alignment and position adapter
Strut
Pallet
Node with receptacles attached

Figure 6. Strut Storage Tray Pallet Details.

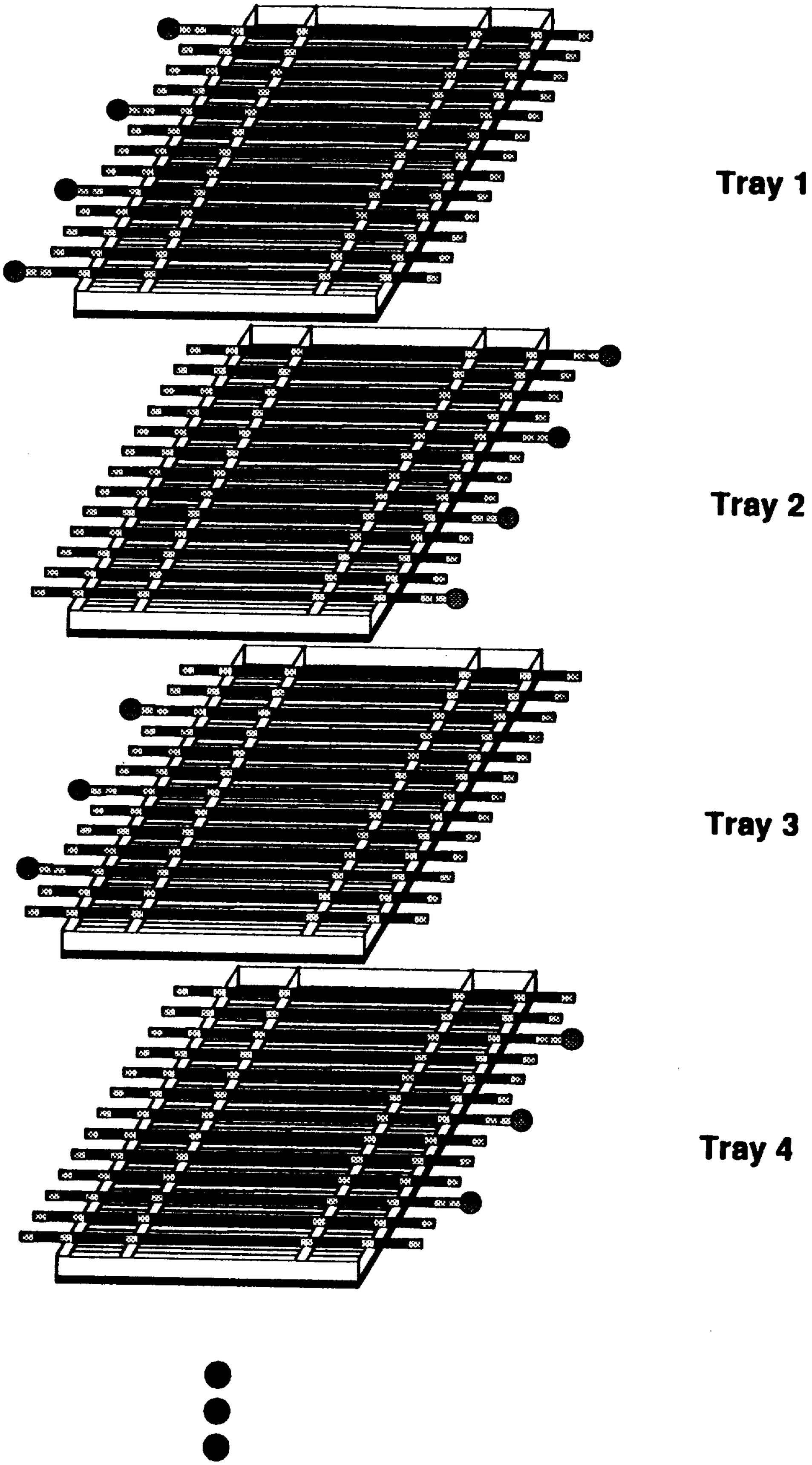


Figure 7. Strut Storage Tray Stacking Arrangement.

AUTOMATED STRUCTURES ASSEMBLY FACILITY

CURRENT CONTROL HIERARCHY

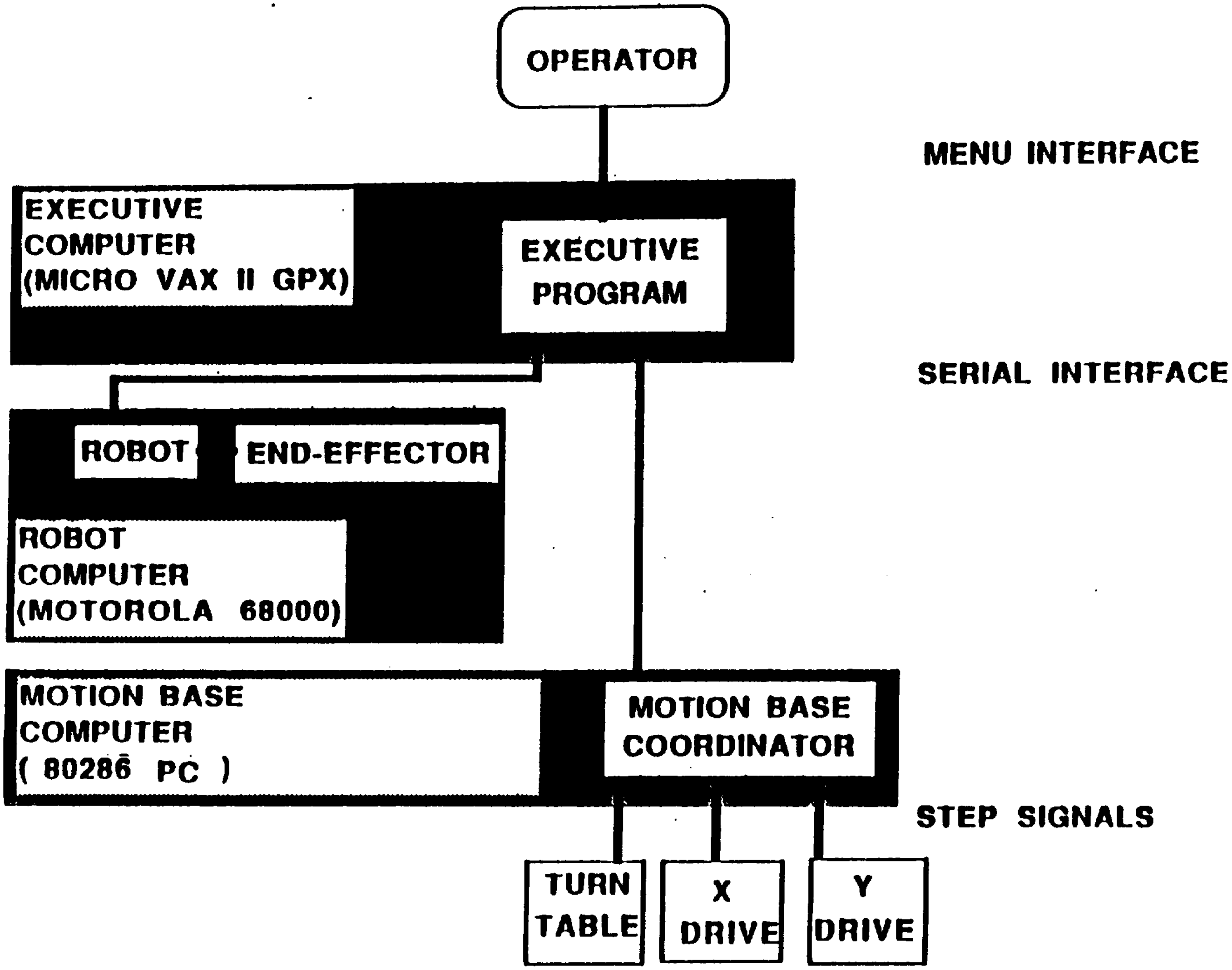


Figure 8. Schematic of Facility Computer Control System.

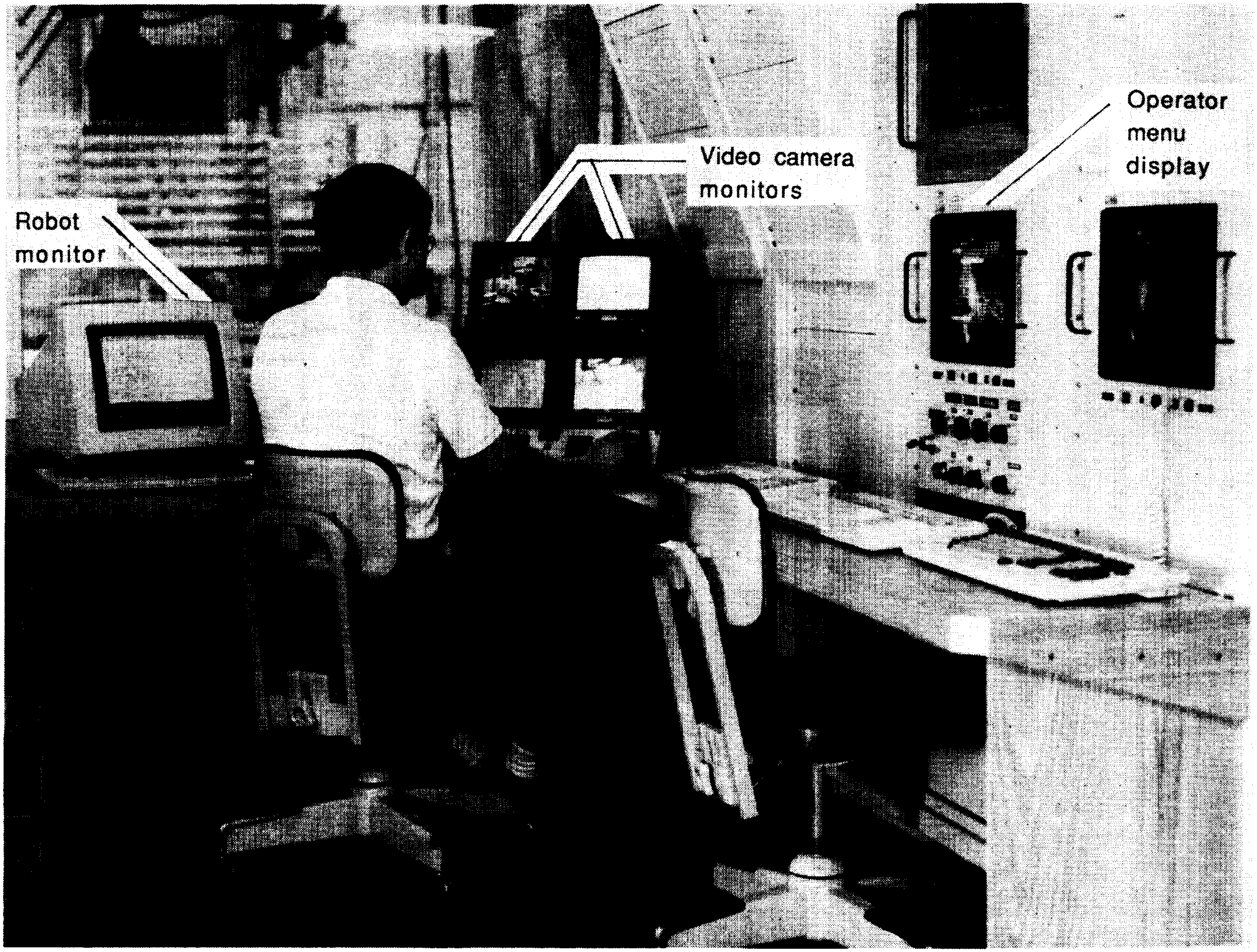


Figure 9. Photograph of Operator's Workstation.

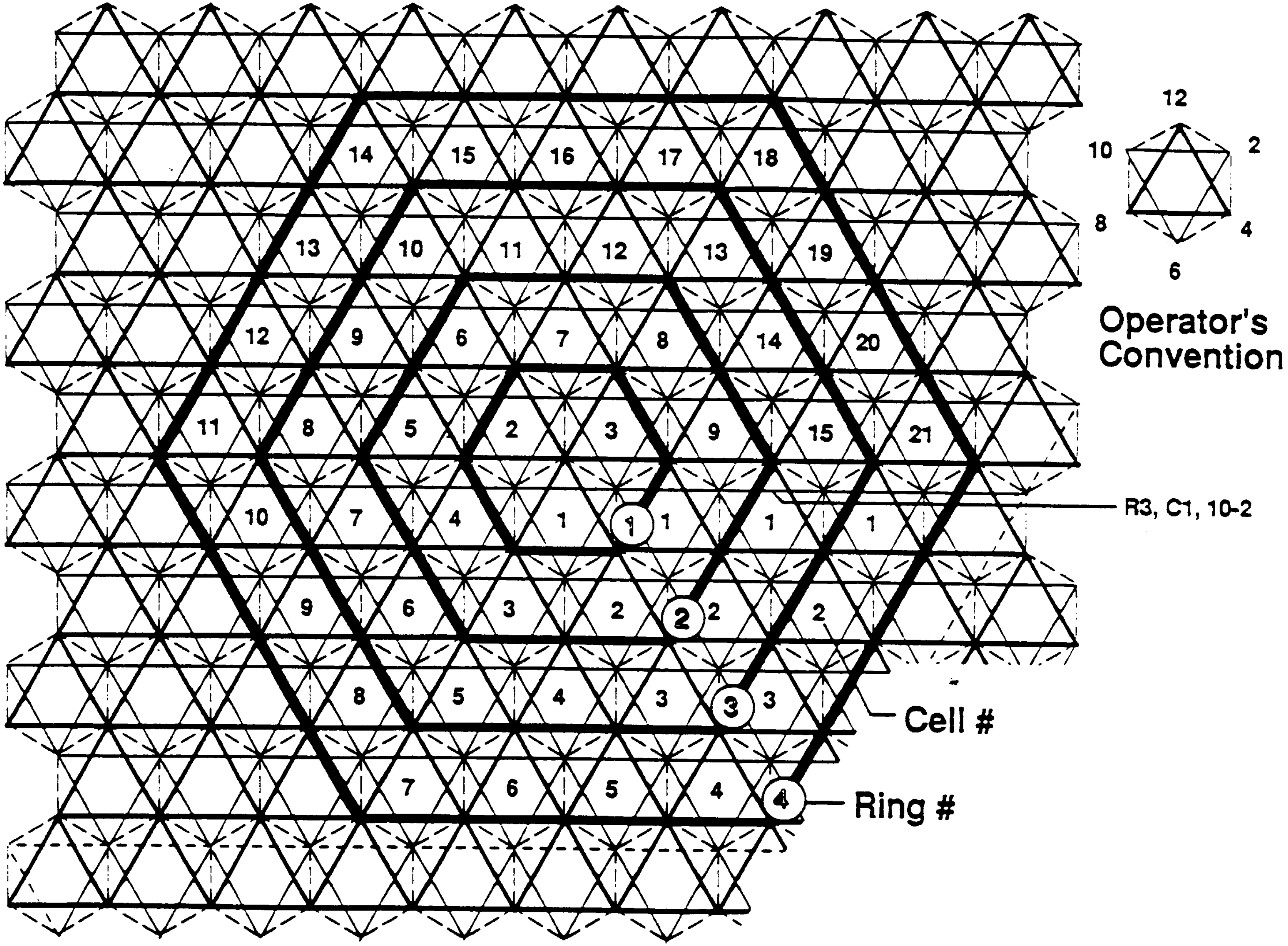


Figure 10. Schematic Illustrating Strut Naming Convention.

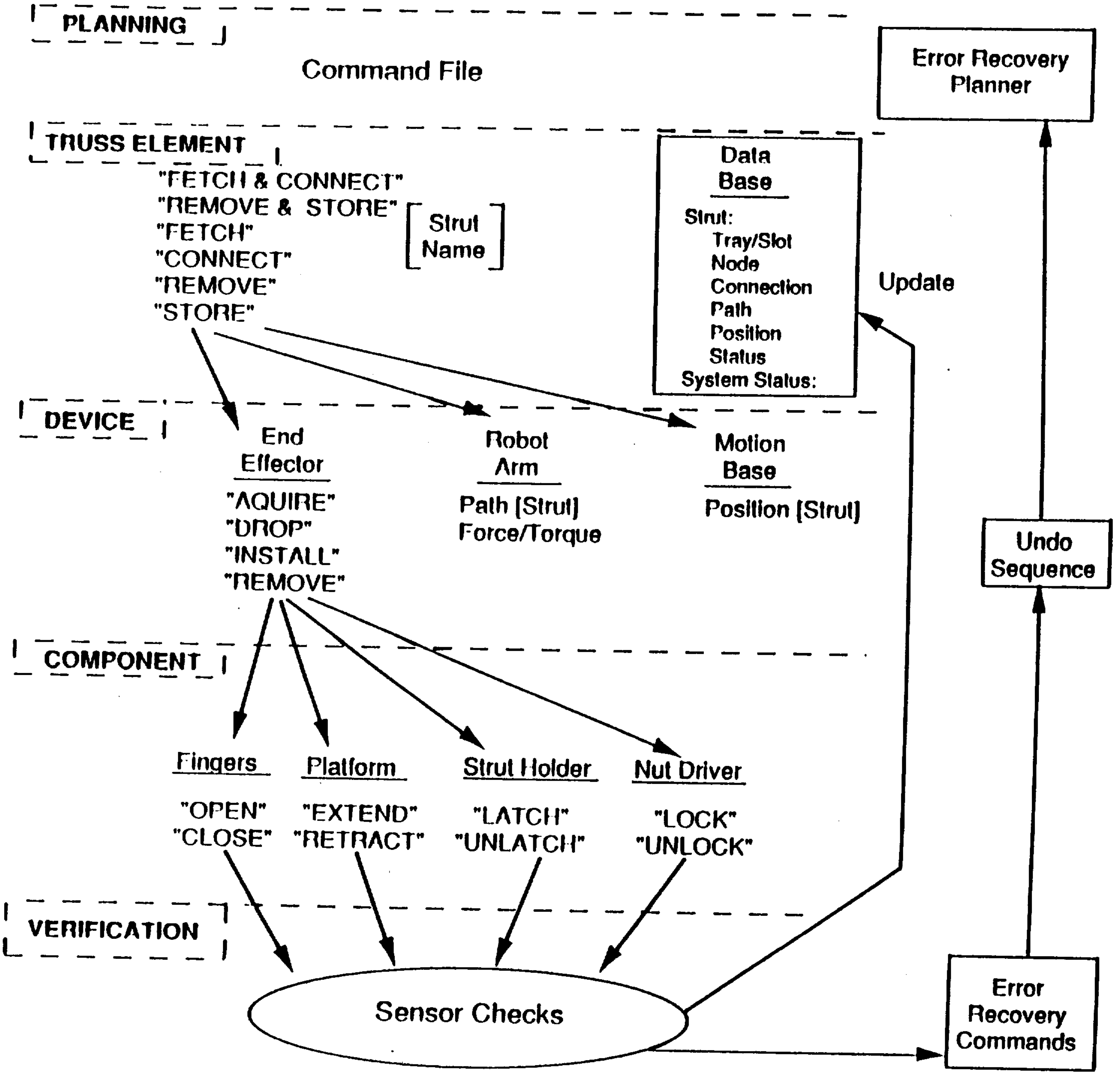
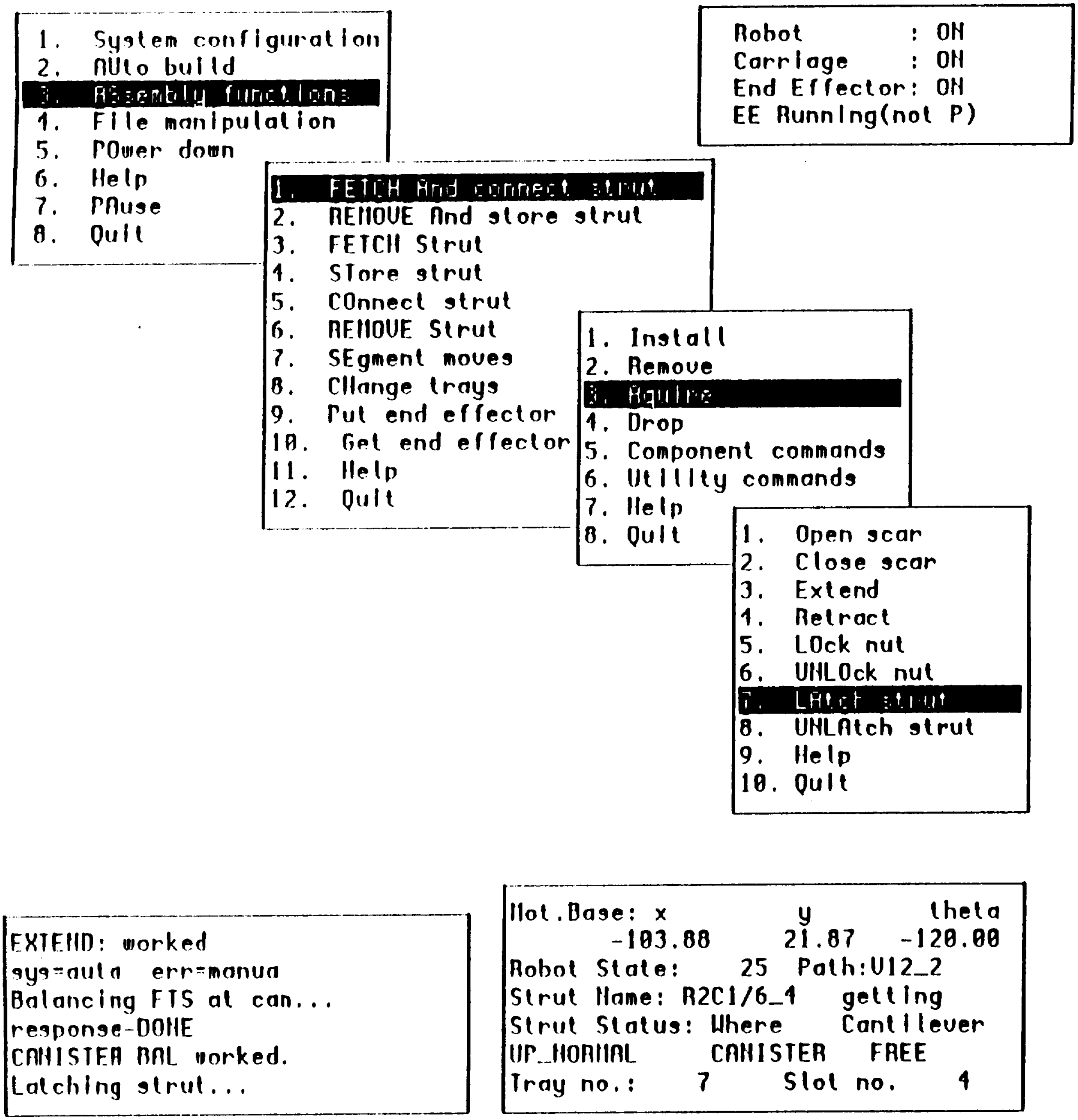


Figure 11. Automated Assembly System Software Hierarchy.



```
Robot        : ON
Carriage     : ON
End Effector: ON
EE Running(not P)
```

```
EXTEND: worked
sys=auto  err=manua
Balancing FTS at can...
response-DONE
CANISTER BAL worked.
Latching strut...
```

```
Mot.Base: x          y        theta
      -103.88     21.87    -120.00
Robot State:     25  Path:U12_2
Strut Name: R2C1/6_4    getting
Strut Status: Where    Cantilever
UP_NORMAL      CANISTER    FREE
Tray no.:     7     Slot no.    4
```

Figure 12. Operator's Menu Display.

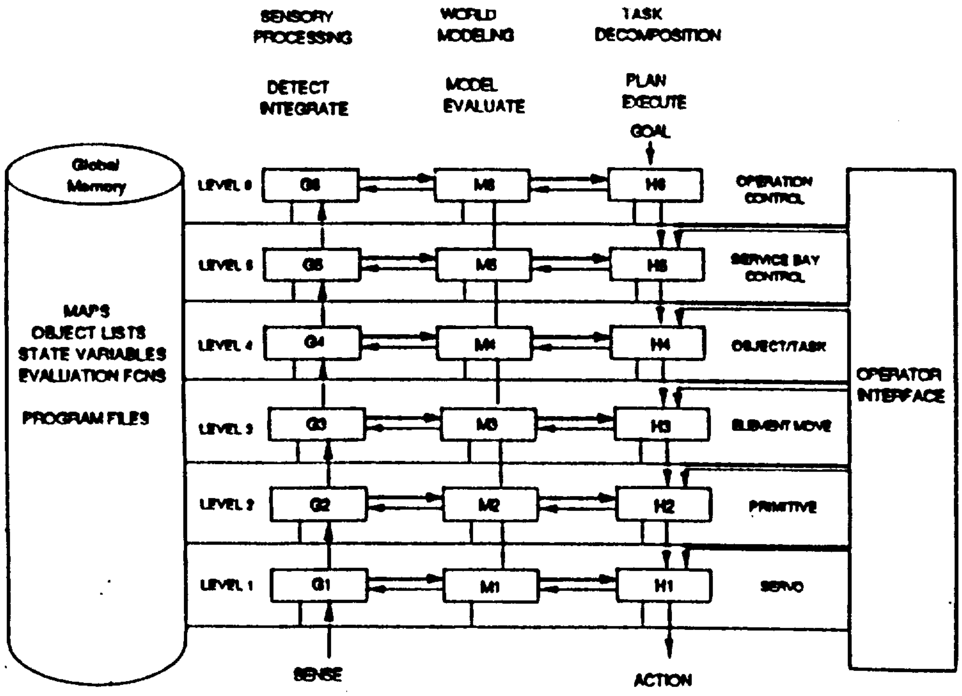


Figure 13(a). NASREM System Architecture.

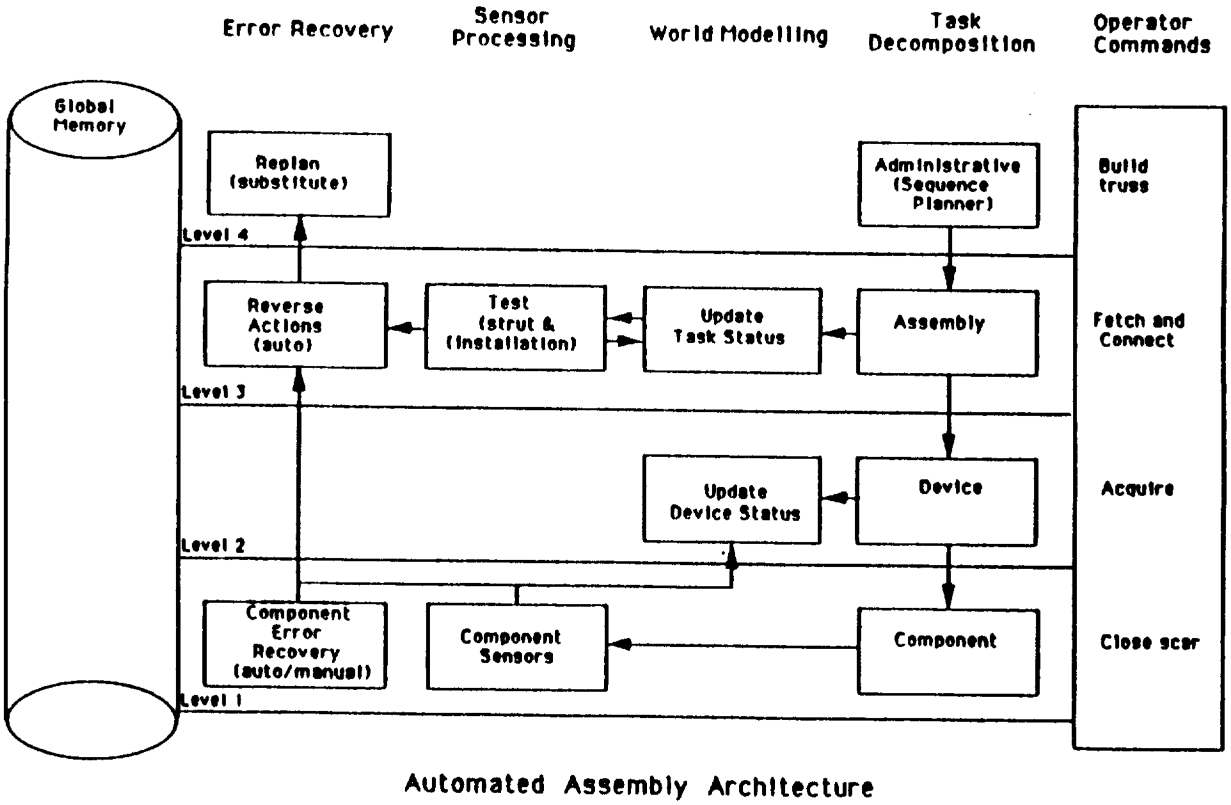


Figure 13(b). Automated Assembly Program Architecture.

Figure 13. Comparison of NASREM and ~~ARTAP~~ ASAL architecture.

Record	Variable	Description
MOTION_BASE_POSITION		Motion base configuration
	X_Car	Carriage x position
	Y_Car	Carriage y position
	Turntable	Theta angle in degrees
STRUT_TYPE		Strut description
	Name	Strut name
	Where	Current strut location
	Connect_To	Struts needed for support
	Loc_In_Cell	Installation position in truss
	Node_End	End with node
	Cap_End	End to capture:
	Cantilever	End conditions of strut
	Flip	Flip indication
	Tray	Tray number containing the strut
	Slot	Position in tray
	State_Pos	Position for each robot state
	X, Y, Z, Roll, Pitch, Yaw	
	X_End, Y_End	Core strut end points for collision avoidance
ROBOT_STATUS		Manipulator arm current mode
	State	Robot path location
	Cond_State	Sub state point in a path
	Strut_Now	Name of strut in arm, if any
	Strut_Getting_Now	Name of strut in process of retrieving
	Strut_Just_Had	Name of last strut installed
TRAY_STATUS		Current tray locations and mode
	Tray_State	Path location
	Tray_Mode	Current tray operation
	Current_Tray	Tray on top
	Working_Ap	Robot position for working approach point
	X, Y, Z, Roll, Pitch, Yaw	
	Storage_Ap	Robot position for storage approach point
	X, Y, Z, Roll, Pitch, Yaw	
TRAY_HANDLE_LOCATIONS		
	Storage_Loc	Robot position in storage canister
	X, Y, Z, Roll, Pitch, Yaw	
	Working_Loc	Robot position in working canister
	X, Y, Z, Roll, Pitch, Yaw	
CURRENT_STRUT		Strut currently in arm
	Left_Seat	Nut driver alignment status
	Right_Seat	
	Left_Nut	Nut status
	Right_Nut	
CURRENT_MOTION_BASE_POSITION		Motion base configuration
	X_Car	Carriage x position
	Y_Car	Carriage y position
	Turntable	Lazy susan theta angle in degrees
END_EFFECTOR		End effector status
	Left_Receptacle_Finger	Left receptacle fingers status
	Right_Receptacle_Finger	Right receptacle fingers status
	Platform	Status of platform
	Latch	Strut holders status
	Storage_Pos	Robot position for storing
	X, Y, Z, Roll, Pitch, Yaw	

Figure 14. Automated Assembly Data Structure.

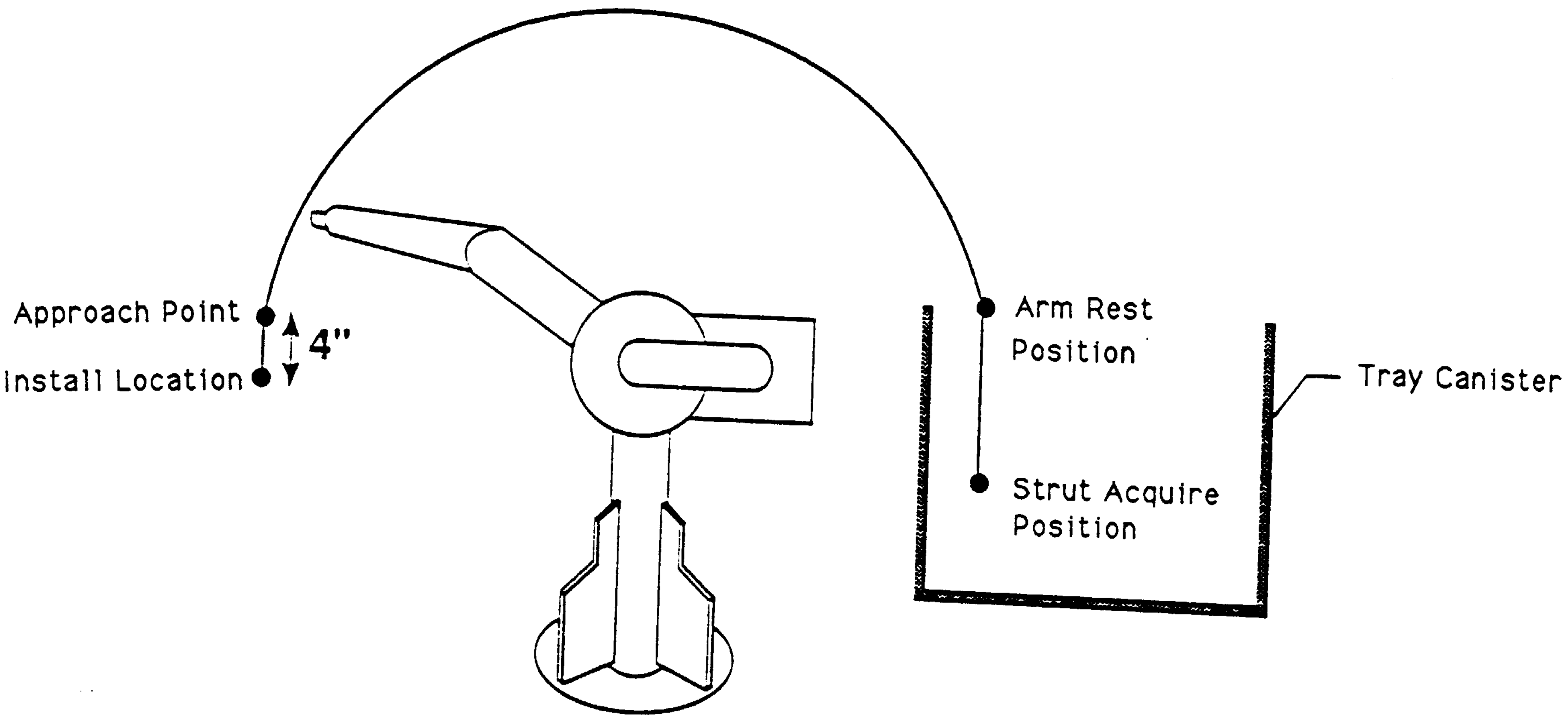
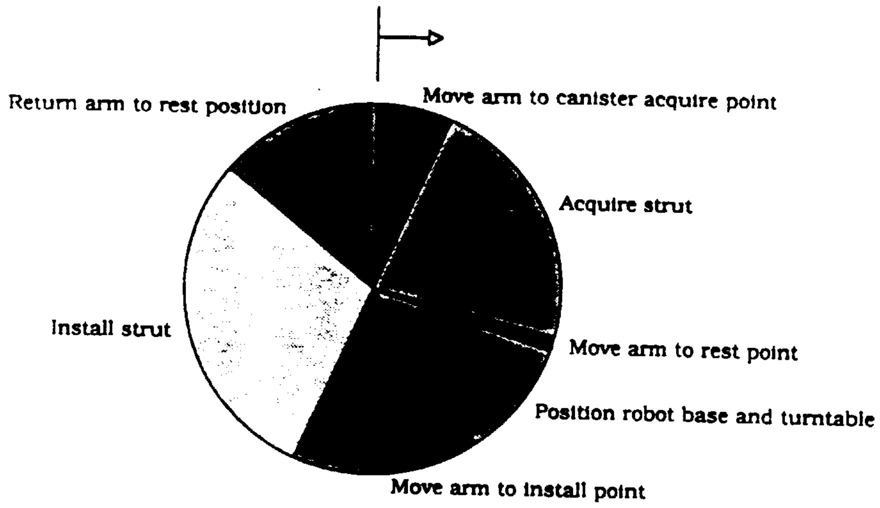


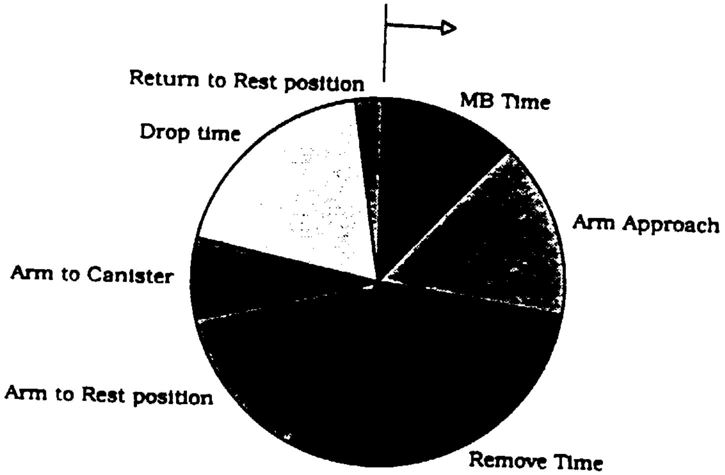
Figure 15. Robot arm path segments.

Total Assembly Time Distribution

NS Assembly 2



Typical Truss Disassembly Time Distribution



	Min.	Percent
Move Arm to Canister	61.42	6.55
Acquire Strut	209.47	22.34
Move Arm to Rest Position	11.50	1.23
Position Motion Base	97.55	10.40
Arm Approach	150.95	16.10
Install Strut	275.78	29.41
Arm Retreat	131.13	13.98
Total:	937.80	

	Minute	Percent
MB Time	107.60	11.83
Arm Approach	142.22	15.64
Remove Time	272.31	29.95
Arm to Rest Position	126.83	13.95
Arm to Canister	64.26	7.07
Drop Time	173.80	19.11
Return to Rest Position	22.28	2.45
	909.29	100.00

Figure 16. Average Time Breakdowns for Complete Truss Assembly and Disassembly.

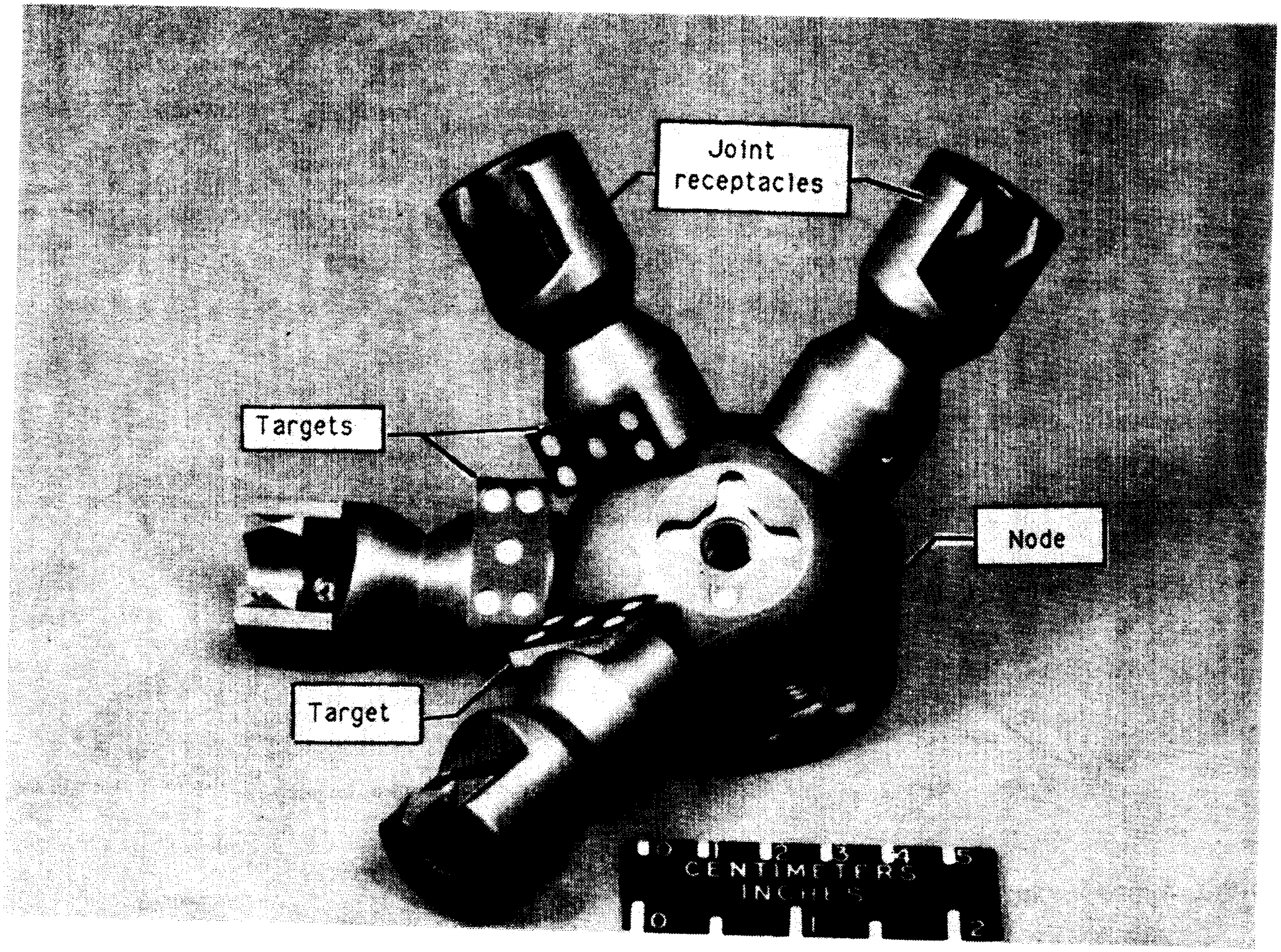


Figure 17. Photograph of Targets for Machine Vision System.

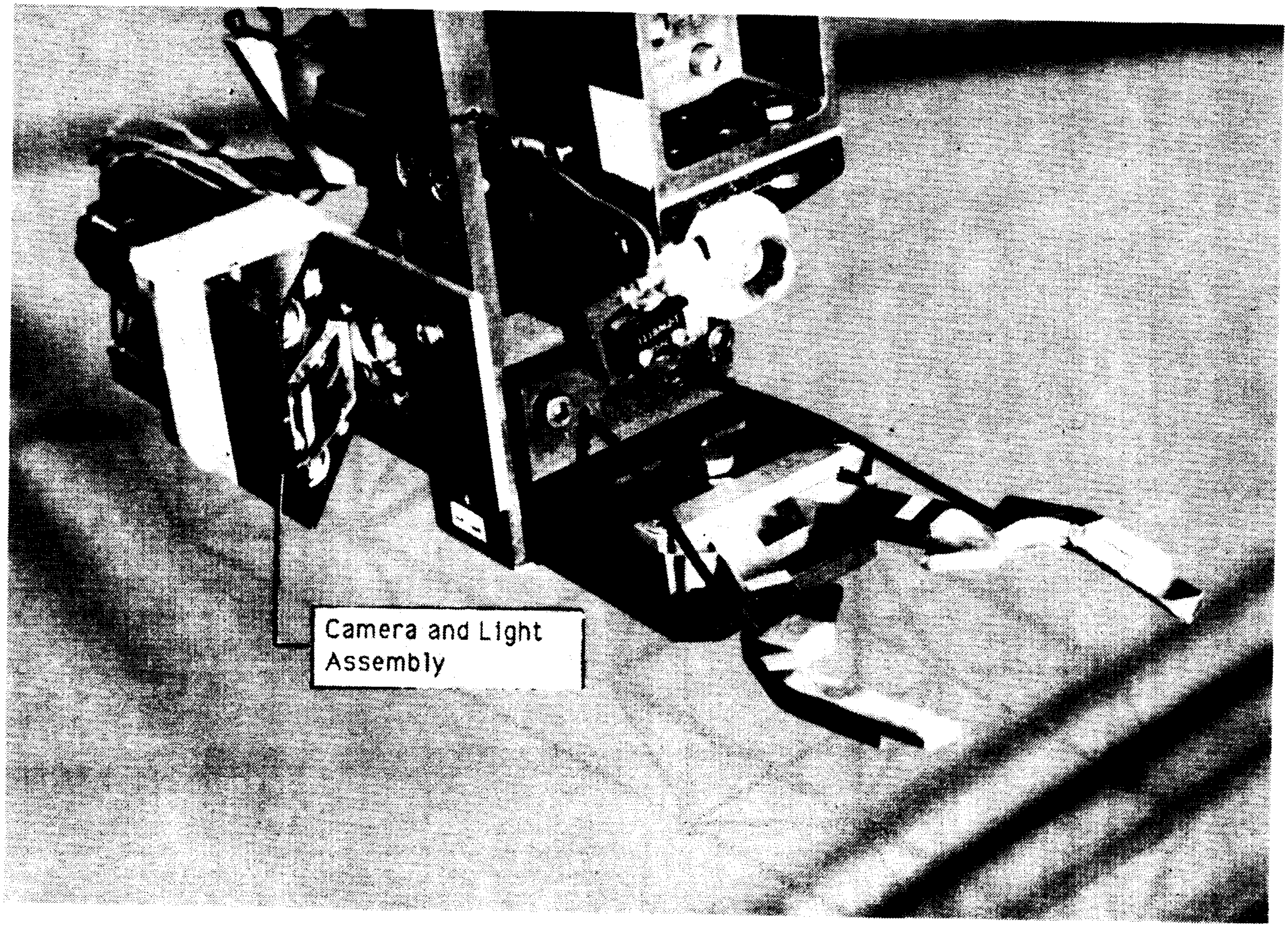


Figure 18. Photograph of Machine Vision System Camera and Lighting.

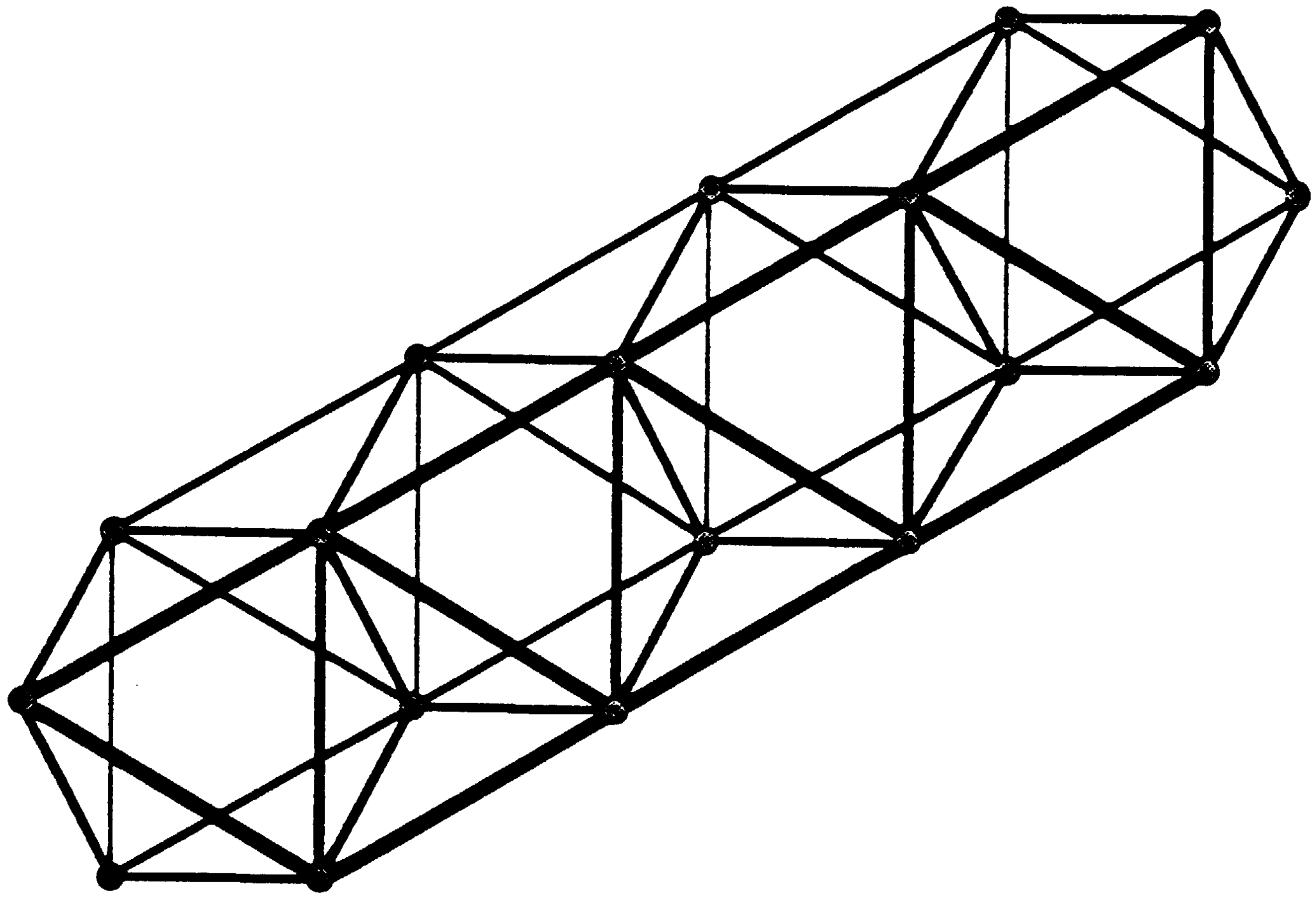


Figure 19. Sketch of Tetrahedral Beam Configuration.

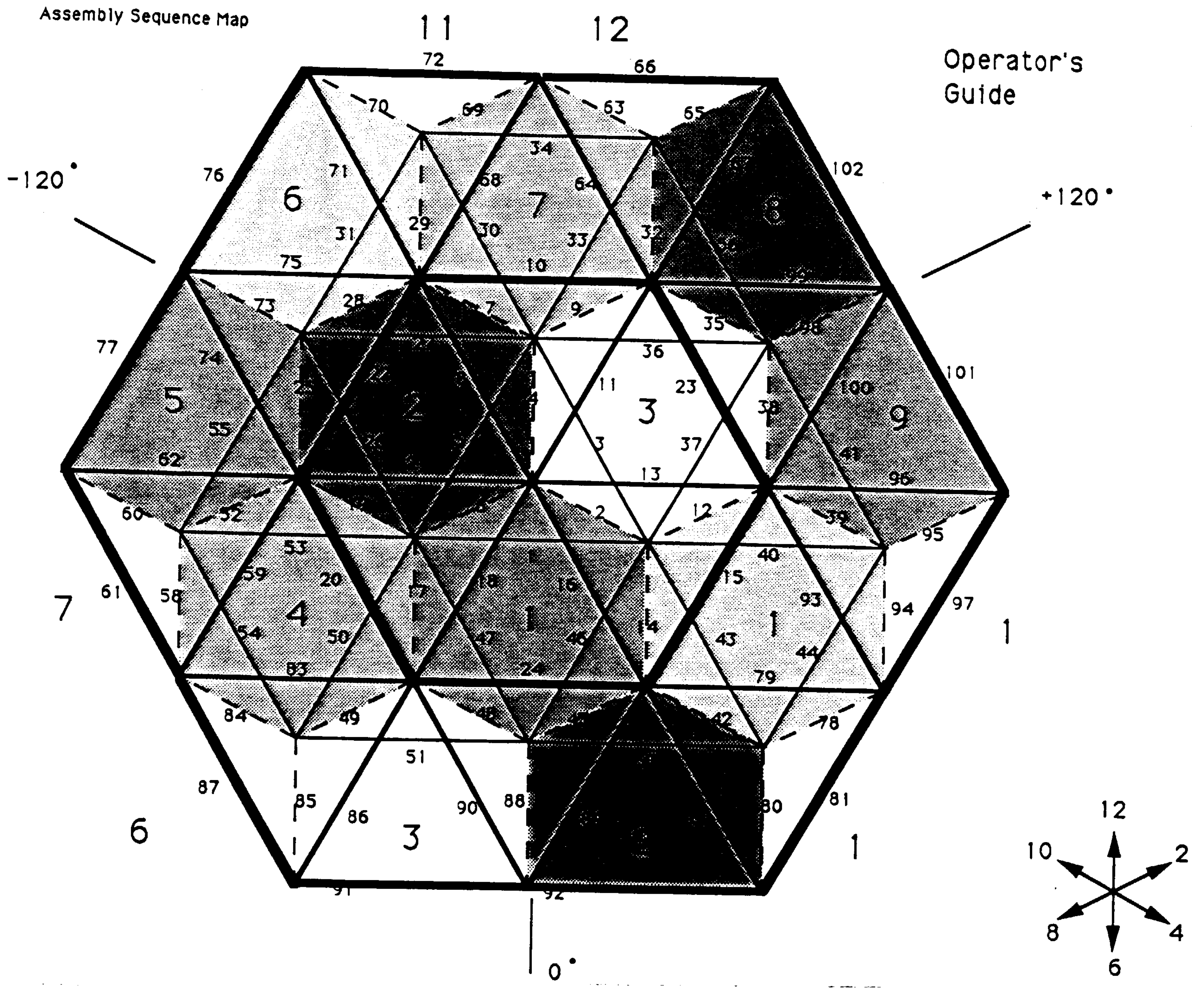


Figure A1. Tetrahedral truss with cells and sequence numbers.

ASAL Carriage Layout

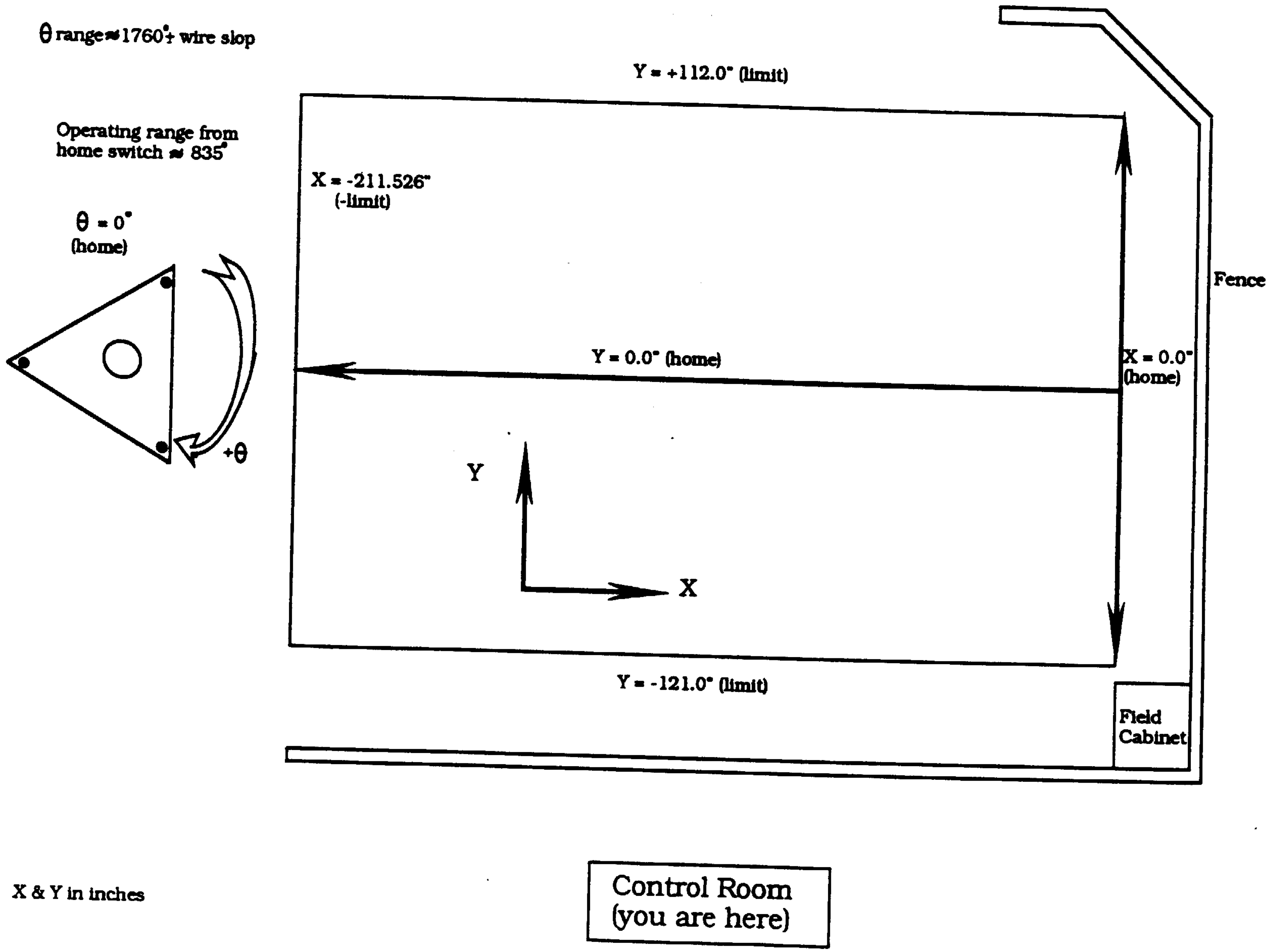


Figure A2. Facility layout diagram.